DOCUMENT RESUME

ED 339 361 IR 015 312

AUTHOR	Sayre, Scott Alan
TITLE	The Development and Evaluation of a Computer-Based System for Managing the Design and Pilot-Testing of Interactive Videodisc Programs. Training and Development Research Center, Project Number Forty-Three.
INSTITUTION	Minnesota Univ., St. Paul. Dept. of Vocational and Technical Education.
PUB DATE	Jun 90
NOTE	281p.
PUB TYPE	Reports - Research/Technical (143) -- Tests/Evaluation Instruments (160)
EDRS PRICE	MF01/PC12 Plus Postage.
DESCRIPTORS	*Authoring Aids (Programing); *Computer Assisted Instruction; Computer Software Development; *Computer System Design; *Hypermedia; Instructional Design; *Interactive Video; Microcomputers; Models; Pilot Projects; Postsecondary Education; Systems Development; *Videodisks
IDENTIFIERS	*Interactive Design Management System

ABSTRACT

The purpose of this study was to develop and validate a computer-based system that would allow interactive video developers to integrate and manage the design components prior to production. These components of an interactive video (IVD) program include visual information in a variety of formats, audio information, and instructional techniques, all of which must be coordinated prior to the production of the actual videodisk. Assumptions were made that users of the program would have prior experience using microcomputers and hypermedia functions, specifically the Macintosh Hypercard computer software system. It was also assumed that users were familiar with computer assisted instruction and interactive video developmental procedures such as storyboarding, scripting, video editing, and computer graphic development. Interactive video design process models were reviewed and modified for the study, and then combined with computer software development and evaluation to arrive at the final Interactive Design Management System (IDMS). The IDMS was evaluated by three interactive video program designers to gauge its developmental utility, efficiency, user-friendliness, and instructional utility. It was concluded that the IDMS works well particularly for novices, but it may be too extensive for many IVD design projects. Appendices include interview questionnaires, a summary of the models reviewed, model documentation, a software manual for the IDMS, and completed evaluation forms. (32 references) (DB)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED339361

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☐ This document has been reproduced as received from the person or organization originating it

☐ Minor changes have been made to improve reproduction quality

- Points of view or opinions stated in this document do not necessarily represent official OERI position or policy

TRAINING AND DEVELOPMENT RESEARCH CENTER

Project Number Forty-Three

THE DEVELOPMENT AND EVALUATION OF A COMPUTER-BASED SYSTEM FOR MANAGING THE DESIGN AND PILOT-TESTING OF INTERACTIVE VIDEODISC PROGRAMS

Scott Alan Sayre

June 1990

university of minnesota
DEPARTMENT OF VOCATIONAL AND TECHNICAL EDUCATION • ST. PAUL, MINNESOTA

Copyright © Scott Alan Sayre

1201S312

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Richard A. Swanson

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

BEST COPY AVAILABLE

Table of Contents

Chapter Page

I. THE PROBLEM

History of the Problem 1
Statement of the Problem 3
Significance of the Problem 4
Purpose and Objectives of the Study 7
Research Questions 8
Limitations of the Study 8
Assumptions of the Study 9
Definition of Terms 10

II. REVIEW OF THE LITERATURE

Systems Theory 14
Computer Software Design and Management 15
Interactive Videodisc Systems 16
Interactive Videodisc Design and Development Requirements.. 17
  Daynes 17
  DeBloois 20
  E-ITV Focus Group 20
  Floyd & Floyd 21
  Iuppa 21
  Johnson, Widerquist, Birdsell, & Miller 22
  Kearsley and Frost 22
  O'Bryan 23
  O'Donnell, Harris, & Sato 23
  Schwier 23

Design Component Interaction 24
Justification for a Comprehensive Model for Software Tool Integration 25
III. DESIGN OF STUDY
Phase 1- Collection of Models from the Literature 27
Phase 2- Collection of Models used by Practitioners 28
  Interview design 28
  Model collection 29
Phase 3- Comprehensive Model Development 29
Phase 4- Software System Development 30
Phase 5- Software Evaluation 31
Experts 32
Reimbursement 32
Information Collection 32
Facilities and Equipment 33
IV. RESULTS OF THE STUDY
Phase 1- Collection of Models from the Literature 34
Phase 2- Collection of Models Used by Practitioners 35
  Model A 36
  Model B 38
  Model C 41
  Model D 44
  Model comparision 49
Phase 3- Comprehensive Model Development 51
  Model summary 51
  Task categorization 52

Final model editing 52
Phase 4- Software System Development 60
Software design 60
Software development 60
Software documentation 62
Phase 5- Software Evaluation 63
Evaluator backgrounds 64
Question 1- Will the system increase work efficiency? 65
Question 2- Will the system increase work accuracy? 65
Question 3- Is the system capable of working within your current work procedure? 65
Question 4- Is the IDMS model complete? 65
Question 5- Is the system complete? 66
Question 6- Does the system offer any unique design and pilot-testing capabilities? 66
Question 7- Would students learning the IV development process benefit from using this system? 66
Question 8- Which tools and accessories did you use? How useful was each tool? 67
Question 9- What is the system's greatest strength? 67
Question 10- What is the system's greatest weakness? 67
Summary 68

V. SUMMARY AND CONCLUSIONS

Evaluation Summary 69
Utility and efficiency 70
Friendliness and instrutional utility 72

Discussion of Additional Findings.......... 73
Recommendations for Further Research and Development.......... 74
Experienced designers.......... 74
Novice designers.......... 75
IDMS software.......... 75
REFERENCES.......... 77

APPENDICES

Appendix A

Practitioner Interview Questions.......... 81

Appendix B

Summary of Models Collected from the Literature.......... 86

Appendix C

Original Practitioner Model Documentation.......... 88

Appendix D

IDMS Software Manual.......... 118

Appendix E

Completed Process Evaluation Forms.......... 224

Appendix F

Completed Summative Evaluation Forms.......... 237

Appendix G

Software Distribution Information.......... 241

List of Figures

Figures

1. Literature Model Summaries.......... 18
2. Software Tool Applications.......... 25
3. Practitioner Model A.......... 37
4. Practitioner Model B.......... 39
5. Practitioner Model C.......... 42
6. Practitioner Model D.......... 45
7. Interactive Design Management System Elements and Definitions.... 53
8. Interactive Design Management System Model.......... 54

CHAPTER I

THE PROBLEM

History of the Problem

In the early days of programmed instruction, instructional designers had the ability to easily analyze and reorganize a program's content because it was primarily text based. Decision points, questions, and answers could be laid out, ordered, simultaneously examined, and reordered without designers losing conceptualization of the resulting instructional program.

As technology continued to advance, automated delivery systems such as computer aided instruction (CAI) began to give designers more complex logic capabilities while simultaneously disguising a program's size and structure in thousands of lines of programmed code. Even though printouts of this code could be read and understood by programmers, paper and pencil flowcharts often were used to simplify a program's operation into generally a more readable form. These flowcharts served a dual purpose of being an outline of a program's structure as well as an index to the pages of text and graphics that composed the program's content. As hardware and software development advanced, user-friendly authoring languages and systems were developed to assist designers and developers in structuring and managing program content. Many of these systems allowed the designers to more easily manipulate instructional programs within the computer by "chunking" or organizing its sequences and variables in a consistent manner. This capability to manipulate the flowchart and its representative text with very little or no reprogramming is quite beneficial because it gives designers the ability to easily view, reorganize, and test the logical presentation of the content once it is input (Huntley & Alessi, 1987). Some of the advanced systems incorporate icon-based authoring environments, in which graphic representations of content elements within a program's flowchart can be cut, pasted, and copied whenever and wherever necessary.

As a result of these advancements in CAI, designers can now integrate the content with the program's logic structure (flowchart) very early in the developmental process. This content integration allows designers to easily manipulate design variables and to pilot test different versions of the same program before arriving at a final design. In the past, the amount of time and work involved in making even simple program changes limited such practices.

Most CAI has been limited to text and graphic information that can be efficiently stored on a magnetic medium. Because of microcomputer's limited memory capacities and processing speed, there has been minimal use of realistic, full-color imagery, recorded sound, and animation. Video motion sequences require such a enormous amount of memory that they are rarely, if ever, stored on magnetic disk.

In early 1970s the first interactive videodisc (IV) system was developed (Daynes, 1984). These systems gave instructional designers the unique ability to integrate audio, video, and computer-generated text and graphics within one interactive delivery system (Braden, 1986). These systems typically consist of a computer system as well as an optical disc that contains still video images and motion sequences. Similar to traditional CAI, IV programs are structured around a sophisticated branching structure that is created with some type of authoring system or possibly written in a programming language. Depending upon the type of interactive program being produced, this programming information may be stored either directly on the optical disc or on a separate magnetic disk (Smith, 1987).

When designing an IV program, instructional designers have a large number of alternative inputs and outputs to choose from. Visual information can consist of text, line drawings, full-color illustrations, animated graphics or cartoons, still video images, motion video sequences, or any combination of the above. Simultaneously, a program's audio can incorporate a single or bilingual narration, soundtrack music, and sound effects. All of these visual and audio sequences are integrated by a computer program that controls the instructional presentation. IV programs also can contain a variety of interactive instructional techniques and options requiring keyboard interaction, graphic interaction, and problem solving, often incorporating level jumping, mastery testing, remediation, elaboration, review and on-line help (Schwier, 1987). The synthesis of all of these capabilities forms an extremely flexible instructional system for presenting information and interacting with users.

Statement of the Problem

A major problem that instructional designers face when designing IV programs is the inability to visualize and pilot test a prototype version of a program before the production of the optical disc (Hawthorne, 1986). Due to all of the different combinations of instructional variables offered by this medium, instructional designers are faced with an almost overwhelming number of options and decisions prior to the actual development process. Currently, the common development process involves a paper-and-pencil method of planning and managing the design of a program (Lippke, 1987).

Separate video sequence storyboards, program flowcharts, narration scripts, production notes, and text and graphic screen diagrams usually are laid out on paper and organized to represent the structure of the completed piece. In some cases, the program's "skeleton" structure and decision points may be input into the computer prior to the production of a work disc in order to check its logic. However, to fully run through, or pilot test a program design involves a complex process of manually cross-referencing separate program and content documents. This complex cross-referencing process makes it very difficult for designers to fully perceive a program's overall image and user friendliness.

At this time, no system has been developed that will allow IV developers to integrate and manage the various design components which compose an IV program prior to production (Cohen, L'allier, & Stewart, 1987). IV designers would greatly benefit from tools that would allow them to integrate and pilot test a program's elements prior to production. Without this ability, designers are restricted to inefficient methods for manipulating and experimenting with the discretely documented structure and variables that make up a program's design. In addition, students who are learning the process cannot fully perceive the complex and integrated nature of the design process.

Significance of the Problem

Without the ability to efficiently manage and integrate the various components of an IV program prior to the production of the disc, designers, and developers (experts and beginners) must rely a great deal on their memories, imaginations, and abilities to speculate the results of the design process. Inaccuracies of design can add significantly to production costs and they can possibly reduce the overall effectiveness of the final product (Floyd & Floyd, 1982).

Careful project management is essential for a cost-effective production. The longer and more complex a program is, the greater the amount of up-front documentation is needed in order to successfully control all production variables throughout the development process. The accurate management of this documentation is even more critical when a program is being produced by a organization or group of people (DeBloois, 1982). In such cases everyone must picture the finished product the same way if it is to be of consistent style and representative of the originally conceived program (Kearsley & Frost, 1985). Production management approaches that are primarily paper based often fail to give the designers the tight, efficient organization to quickly access all of the production information related to a specific section of a program.

In addition, documentation updating and program changes are often time-consuming and inaccurate since one change may effect many other elements of the program. For example, the replacement of a video sequence with a still frame may also require changes in scripting, timing, and programming logic. If changes are not carefully managed, other elements that are effected by the change may go unnoticed until they become a problem later in the production process or, even worse, remain unseen until the program has been completed. Constant, careful analysis and consideration of all program changes is essential for accurate management of design process (Daynes, 1987). Cross-referenced paper-based management systems greatly restrict aperson's ability to do so.

Due to the complexity of paper-based level IV design approaches, designers are also quite restricted by their inability to integrate the components of a program until its actual production. This restriction limits the designers ability to evaluate and experiment with different instructional techniques and designs before the production of a work disc (Hawthorne, 1986). This restriction is very unfortunate because this type of pilot testing can be and often is done with other types of instructional delivery systems. And with all the different combinations of instructional variables IV has to offer, the opportunity to evaluate their effects is much needed (Braden, 1986). Currently, without integrated design tools, program pilot testing can be done only with the crude cross-referencing technique, or by waiting until a work disc is produced. By the time a work disc has been produced, the project has often progressed too far to make any major conceptual changes without severe financial losses.

Finally, one other context in which the inability to integrate and pilot test the components of an interactive program poses many restrictions is in the classroom. Students learning the methods of IV program design are often limited to developing paper-based models or writing programs for preexisting optical discs. Although teaching with the paper-based program design method has the advantage of allowing students to incorporate every possible IV instructional technique, it is subject to all of the pilot-testing and managing limitations discussed above. The other option of having students write an instructional program for a preexisting optical disc offers them the opportunity to actually interact with the disc as well as test their program during and after its development. However, the material and techniques a student has available during both the design and development stages is very limited by the material recorded on the optical disc.

Purpose and Objectives of the Study

The purpose of this study is to develop and validate a computer-based system for managing the design and pilot testing of IV programs.

The objectives of this study will be to:

(1) Identify a set of project elements and tasks necessary for designing IV programs.

(2) Develop a unique model of the project elements and tasks involved in the design of IV programs.

(3) Develop a system to allow IV designers and developers to design and efficiently manage and integrate the various project elements and tasks involved in the IV production process.

(4) Develop a software program that will allow the simultaneous integration of all program elements in order to facilitate the manipulation and pilot testing of new programs or program sequences.

(5) Evaluate the utility and efficiency of the completed system in designing and pilot testing interactive videodisc programs.

(6) Evaluate the user friendliness and instructional utility of the completed system.

Research Questions

The development and evaluation objrctives of this study will be directed to answering the following questions:

(1) What are the common elements and tasks involved in the IV design process?

(2) What are the different types of documentation used in the interactive video development process?

(3) Will IV designers and developers find this system to be an efficient and accurate means of managing and pilot testing an IV program's elements prior to production?

(4) Will this type of software system to be an effective tool for teaching beginning IV designers how to design and test pilot test programs?

Limitations of the Study

The following limitations were present in this study:

(1) The software's pilot testing capabilities will be evaluated only on their ability to facilitate the review of alternative interactions between program elements rather than simulating the operation of an entire program.

(2) The resulting software will be run only on Macintosh computers which have at least one megabyte of memory and a 20 megabyte hard disk.

(3) Video sequences containing motion and complex computer animation will be represented only by a set of storyboard illustrations rather than the actual sequence that will appear in the final production.

(4) The programming and management options offered by this system will represent only the standard features of common IV authoring systems.

(5) All text screens, video images, and illustrations will be limited to black and white.

(6) All audio sequences, including narrations and soundtracks, will be represented as text screens and production notes because the system will not be capable of reproducing "live" audio sequences.

Assumptions of the Study

Two assumptions were made during the development of the system. The first is that all of the users will have had previous experience with the Macintosh computer and the basic Hypercard operating environments. Because of this, no instructions were presented to the user about how to operate the computer or perform basic browsing, typing, and selecting functions within Hypercard.

The second assumption is that the user had a basic understanding of the operation of an IV program and the basic developmental procedures within its production. This understanding includes experience in the areas of storyboarding, scripting, video and audio editing, computer graphic development, and background in different interactive programming options.

Definition of Terms

The definition of the technical terms contained within this study are as follows:

Authoring system:	A computer program which assists computer program authors by simplifying the computer program writing process.
Branching:	The ability of the computer program to redirect individual users according to their input.
Flowcharting:	The visual depiction of the decision-making processes within a computer program.
Full-color	An attempt to reproduce the colors of a scene or object as it appears in the real world, usually requiring atleast 256 colors on a computer.
Graphic screens:	A computer or video screen containing still images or illustrations rather than text or motion video.
Hypercard:	A unique computer program authoring system that simplifies the program authoring process and runs other program "stacks" developed with Hypercard.
Icon based:	Any computer program or system whereby the user interacts with and manipulates graphic icons rather than using the keyboard.
Integrated software tools:	A software system that incorporates a wide variety of applications and is capable of sharing and updating information between the applications.

Interactive:	Any computer or video program that requires user input or control and generates some form of feedback.
Level 3:	An advanced type of IV program whereby a changeable, external computer program controls the videodisc player and user interaction.
Magnetic disk:	An erasable hard or floppy disk which is used to store and retrieve computer data.
Mastering:	The step in the videodisc production process in which a videodisc is produced from which duplicates will be made.
Optical disc:	A plastic coated aluminum disc that is used as a permanent medium for storing audio, video, or computer information.
Pilot testing:	A trial run of a IV program before disc mastering and duplication.
Production:	The process of recording and editing audio and video information for use in video programs. Also the generation of computer programs and graphics for use in IV programs.
Program design:	The process of identifying and outlining the key components and treatment of an IV program.
Program development:	The process of writing, scripting, flowcharting, and storyboarding an IV program.
Prototype:	A trial production of an IV program before disc mastering and duplication.

Storyboarding:	The visual depiction of visual information that will be contained in a finished IV program.
User-friendly:	A computer program that is easily understood and manipulated by the user.
Videodisc:	An optical disc containing audio, video, and ,sometimes, computer information.
Work disc:	An inexpensive, low quality, one-of-a-kind videodisc that can be used to pilot test an IV program before mastering.

CHAPTER II

REVIEW OF THE LITERATURE

This review of literature discusses the basic aspects of system theory and its relationship to the processes involved in interactive videodisc (IV) program design and development. The discussion is divided into five major sections, beginning with the foundations of the systems approach to design and analysis. The second section discusses the implications of systems theory as it relates to computer software design and management. Third, the discussion introduces interactive videodisc systems and their subsystems. The fourth section discusses the design requirements of each of the subsystems during the IV program design process. The fifth section discusses the interactions between these subsystems and the ways in which software tools could aid in their management during a program's design stage and pilot testing. The sixth and final section focuses on the literature's justification for a comprehensive design model for software tool integration. Through this review, IV's unique design management needs will be made visible.

The study of IV program design and production is the study of the various processes involved during each stage of a programs development. Similar to most other instructional technologies, IV programs are built upon a large number of interdisciplinary foundations. In order to fully understand the complex interactions between these diverse disciplines, a systems approach is necessary (Banathy, 1987).

Systems Theory

The coming of the information age has brought with it many new challenges requiring a rethinking of previously acceptable design and analysis techniques. Over time, it was found that the analytical reduction methodologies of the classical sciences failed to explain the complex interactions within many phenomena. Although the analytical approach worked well for examining certain types of phenomena, it was dependent upon two conditions. First, it is restricted to processes whereby there is no significant interaction between components. Thus, each component must exist in the same form independent of its role within a process. Second, the relationship between the components must be organized linearly within a process (Bertalanffy, 1968). These conditions limit number and types of interrelationships that are possible between components. Many phenomena cannot be accurately represented linearly or in a summative form (Laszlo, 1972). Because of reductionism's inherent deficiencies, many holistic approaches to analysis began to emerge.

One of the holistic design and analysis approaches that has gained wide acceptance is systems theory. The systems approach avoided the restrictions of the analytical process by identifying principles for describing relationships between system components. Sophisticated technologies such as space travel and nuclear power generation made the systems approach a necessity. These technologies are constructed by teams of specialists from elements of many diverse scientific disciplines (Aguilar, 1973). Because each discipline has its own specialized rules, a more general set of principles had to be adopted in order for the team to analyze or design an entire system (Bertalanffy, 1968). Once the relationship between the various components and subsystems within a complex system are defined, system models may be constructed to illustrate systems and their interactions. Three different types of system models are available. The systems-context model examines systems relationships within their environments. Structural models define the way a system is organized. Process models describe how systems operate and how they change over time (Banathy, 1987) . These models provide research and design teams with common language by which they can communicate.

Computer Software Design and Management

Systems theory became widely accepted in the computer science area because of its ability to describe the complex interaction between system and software components. Conner (1981) explained, "The problem of system development is one of establishing uniformity, discipline, and structure to the process of planning, analyzing, designing, implementing, and maintaining computer based systems" (p. 216). The linearity of the English language made it very difficult to accurately describe nonlinear software interactions. However, graphic models can visually depict all aspects of even the most complex system in a greatly simplified form (Chapin, 1981).

During the software design process, designers must identify and organize program components into a an integrated system. This process involves two progressive steps. The first step, abstraction, involves the identification of major program features and structural elements. The second step, elaboration, requires the designer to break the program into smaller, manageable components and specify the relationship between them (Baker, 1981). Once software design has progressed to the elaborative step, the amount and complexity of the components involved often become too extensive to manage with manual paper-and-pencil, methods. In order to overcome these information management problems, computer-based systems should be used to construct dynamic system models. These system models gave software designers a way in which to accurately communicate their designs to other members of a development team. System components can be fully integrated during the design process so that each is designed to work with the whole system rather than independently (Konsynski, 1981). Regarding systems technology, designers have the ability to operationalize their concepts and develop integrated software solutions (Laszlo, 1972). This type of dynamic system modeling can be applied to any entity once it is described as a system.

Interactive Videodisc Systems

Interactive videodisc systems are one of the most complex instructional delivery systems available. Like many other advanced technologies, the development of interactive video was the product of research in many different disciplines. IV is a sophisticated computer-based system. Although IV systems have many of the same characteristics and components of computer systems, their design is further complicated by the integration of a videodisc system. In addition to these two electronic technologies, IV systems also have theoretical foundations in the areas of instructional and visual design (Braden, 1986). Because of these broad foundations, the design of interactive videodisc programs requires expertise in a variety of areas including, learning theory, instructional design, computer programming, video production, and graphic design (Smith, 1987). Each of these specialized areas of IV program design has it own unique systems and requirements. Each could area could work independently. But, an IV program is a specialized system. In order for an IV instructional program to be effective, all of its subsystems must be considered and integrated during all phases of the design process. A team of designers who have different areas of expertise should work together in order to optimize this process (Kearsley & Frost, 1985). To ensure that every aspect of IV program design and development process is considered and controlled for, teams often adopt some type of guidelines or process model.

Interactive Videodisc Design and Development Requirements

The subsystems and components of an IV program must be identified and and carefully designed to assure the quality of the finished product. Many books, articles, and models have been published that describe different approaches to program design and development. Figure 1 contains an interpreted, categorized summary of 10 of the most in-depth publications that describe IV program design and development.

IV is a medium that has broad applications in education and training, as well as sales and entertainment (O'Donnell, Harris, & Sato, 1981). Because of diversity of IV's applications, design approaches often differ in emphasis. In order to fully understand the varying breadth and intent of each developmental approach, the context of each description must be considered before studying each model in detail. The context can be clarified by considering the description's source, and emphasis and the author's background.

Daynes. Daynes (1987) has be credited with designing and producing more IV programs than any other person. His design procedure was extracted from the course notes from one of his comprehensive workshops on IV program

Literature Model Summaries

AUTHOR	FRONT END ANALYSIS	CONTENT OUTLINE	PROGRAM DESIGN	PRODUCTION REQUIREMENTS	PROJECT MANAGEMENT
Daynes, R. (1987)	Define Budget Define instructional goals Analyze population Define hardware Define software Assess system capabilities	Identify SMEs Define ternimal objectives Define enabling objectives Conduct task analyses Design learning hierarchies Develop test items Lesson specification Check content vs. objectives	Define level of interactivity Define instructional strategies Select appropriate media Design individual lesson flow Define types of interactivity Define user interface Define input devices	Define creative treatments Develop scripts Create visual notebook Develop documentation Select talent Define production sites	Identify design team Time management plan Produce budgeting plan Design team responsibilities Cont. review procedures Production schedule
DeBloois M. (1982)	Describe problem Needs analysis Determine funding Time requirements Personnel requirements Specific goals Define different Learner charactersistics Learning environment Internal evaluation criteria External evaluation criteria Hardware restrictions Software restrictions	Task analysis Learning hierarchies Content structure/analysis SME content report Produce script SME content review Test items and location	Select media for objectives Define cues	Define treatment and style Audio script Video script Computer script	Sytematic plan of action Get production bids
EITV Focus Group Hawthorne, & Lippke (Eds.) (1986-87)	Disc concept Identify SMEs Needs analysis Terminal objectives Enabling objectives Cost benifit analysis	Task analysis Produce script	Level of interactivity Flowchart Audio track usage Audio style consistency Video style consistency	Review existing media Describe treatments Storyboarding	Assign responsibilities Project management outline Team communications Content review strategy
Floyd, S. & Floyd, B. (1982)	Define problem Behavioral objectives Learner characteristics Estimate costs	Task analysis Structure curriculum chunks Identify remedial areas Review with SME Scripting	Instructional strategies Evaluation strategies Outline program Flowchart Branching techniques Audio transition consistency Video transition consistency	Storyboard	Itemized budget Pilot testing
Iuppa, N. (1984)	Define problem Learner characteristics Special learner problems Behavioral objectives Identify discriminators	Identify SMEs Task analysis Methods of evaluation Script development	Types of interactivity Instructional strategies Introduction system User control method Practice methods Flowchart	Storyboard	

Figure 1. Literature model summaries

Literature Model Summaries

AUTHOR	FRONT END ANALYSIS	CONTENT OUTLINE	PROGRAM DESIGN	PRODUCTION REQUIREMENTS	PROJECT MANAGEMENT
Johnson, J. Widerquist, K Birdsell, J. & Miller, A. (1985)	Define needs Define goals Behavioral objectives	Content outline Learning hierarchy Chapter hierarchy Produce script	Level of interactivity Input devices Team storyboard Flowchart Branching	Videodisc production sheet Computer production sheet	Indiv. formative evaluation Group formative evaluation Trial summative evaluation
Kearley, G. & Frost, J. (1985)	Time limitations Budget limitations Learner characteristics Learners work environment Instructional objectives Hardware limitations Software limitations	Learning hierarchy	Input devices Instructional strategies Level of interactivity Types of interactivity Personality of disc Audio transition consistency Video transition consistency Integration of other media User control options		Design team coordination Disc space management Budget keeping
O'Bryan, K. (1982)	Needs identification Audience characteristics Instructional objectives Assess system capabilities	Task analysis Instructional modules Test items and locations Produce script	Instructional strategies Types of interactivity Flowchart Branching	Storyboard	Evaluation strategies
O'Donnell, J Harris, D. & Sato, T. (1981)	Needs assessment Program objectives Define audiences Learner characteristics	Content for each audience Design tests Produce script	Types of interactivity Define control segments Audio track usage Flowchart	Storyboard	
Schwier, R., (1987)	Define audiences Audience characteristics Audience size Locations of audiences Learning environments Organizational commitment Available budget Time limitations Behavioral objectives Instructional objectives	Identify SMEs General content outline Core instructional elements	Level of interactivity Select appropriate media Instructional strategies User control design Audio track usage Flowchart Branching strategies Audio transition consistency Video transition consistency	Technical standard spec. Instructional format spec. Aesthetic specifications	

Figure 1. Literature model summaries

production. His approach is that of a client-centered, external IV program design and production organization. Daynes described the program design and development process as being carried out by a interdisciplinary design team working directly with the client. A hierarchical methodology is used to develop program content. Daynes was founder and director of the Nebraska Videodisc Design/Production Group. Although he maintains an approach suited to commercial productions, he has produced IV programs for a wide variety of educational and corporate applications.

DeBloois. DeBloois, (1982) offers an in-depth approach to IV design, development, and production for both educational and business applications. The overall process is described from a system viewpoint. DeBloois stresses pedagogical considerations unique to IV in its ability to conform to individual learning styles and characteristics. Formative and summative evaluation of program development is also emphasized as well as the careful management of time and resources. Much of his research was derived from a 1979 U.S. Army IV application and his work at Utah State University.

E-ITV Focus Group. The Educational and Industrial Television Interactive Videodisc Focus Group approach was extracted from a series of five articles edited by Hawthorne (1986) and Lippke (1987). This incomplete series of articles resulted from a series of focus interviews with 10 key authorities in the design, development and production of IV programs. This group included Rod Daynes, Michael DeBloois, and Steve Floyd, each of whom has published independent approaches that are also included within this study. The interviews were carried out using modified focus group and Delphi techniques. Because of this approach, none of the statements could be attributed to a specific source, and conflicting opinions were often expressed.

Most of the comments were from the position of an external developer working with a subject matter expert (SME). The group stressed a two-step design process to increase budgeting accuracy. The process consisted first of a proposal to management. Once accepted, the specific program content is designed. Most participants also agreed upon a team approach to IV design, although careful team management was also stressed.

Floyd & Floyd. Steve Floyd and Beth Floyd (1982) describe the design process from the standpoint of a media production department within a large corporation. Although IV design is described as an internal process, much consideration is given to proposing IV as a solution to upper management. The importance of conducting a cost/benefit analysis and focusing on managerial objectives is emphasized during the proposal stage. IV program design and production steps are outlined as a means for organizing its budget and formatively evaluating its quality. Three different teams are described in the development process, including a design team, a creative team, and a production team. Many of these corporate concerns result from Steve Floyd's extensive work as videodisc developer within Coca Cola and Texas Instruments and Beth Floyd's work as an independent video producer.

Iuppa. Iuppa's (1984) book dedicated strictly to the design of IV programs approaches the entire process from an external developers standpoint. He also describes a two-part design document consisting of a client proposal and a design outline for developers. The importance of detailed analysis procedures and the creative design of highly interactive treatments are emphasized. Iuppa describes design considerations from a programs conception all the way through its final production. Assistance of SMEs is described during the content design phase: however, a "design team approach" is not addressed. Iuppa's focus on the design of IV programs for training results from his previous work as an instructional designer for Bank of America. He has also worked on developing IV shopping systems.

Johnson, Widerquist, Birdsell, and Miller. Johnson, Widerquist, Birdsell, and Miller's (1985) article focuses primarily on storyboarding IV program designs. The article is based on the development of a science disc for a secondary school. Although the authors recommend a preliminary design and content analysis, their main emphasis is on a three-stage storyboarding process. These stages consist of a group storyboarding session, a formative evaluation procedure, and the generation of detailed production sheets. Unlike standard storyboarding processes, their procedure involves a group brainstorming session in which the treatment of the content is developed. They recommend that individual and group formative evaluations be conducted with the storyboarded content before disc production details are specified. All four of the authors are in involved in instructional design and IV development at the University of Notre Dame.

Kearsley and Frost. Kearsley and Frost's (1985) article summarizes some of the design concepts that have been found to be important when developing IV programs. Although the authors do not present a complete procedure for IV design, they describe significant instructional and visual research findings that should be considered during the design process. Use of interesting instructional techniques and high quality visuals is stressed. The authors also address the importance of disc organization and the need for a team approach to program development. Both of the authors work as instructional designers and producers for independent media production companies.

O'Bryan. O' Bryan (1982) approaches IV program development from an educational standpoint. He emphasizes the importance of instructional design throughout the entire development process. O'Bryan stresses that IV's strength is its ability to avoid the restrictions of conventional television. He encourages the application of learning principles and consideration of different learning styles during the design process. Integration of IV's unique data-gathering capabilities is also discussed as an excellent means to monitor student progress and to evaluate program effectiveness. O'Bryan's educational interests in IV results from his work with the Children's Television Workshop and his extensive background in educational television management, production, writing, and research.

O'Donnell, Harris, and Sato. O'Donnell, Harris, and Sato's (1981) unpublished booklet describes a design approach for novices who are interested in designing and producing IV programs for commercial and corporate use. Differing from most design approaches, the authors describe interactivity as a method for allowing different audiences access to different levels of information. They recommend this type of design for multilevel training and marketing applications. Because this article's primary focus is on the design of level II IVs, interactivity and programing options are limited since no external microcomputer can be used. All three of the authors work within Sony's Communication Group which produced this booklet.

Schwier. Schwier's (1987) approach to IV design was extracted from his in-depth book on IV foundations and development. Although this book focused primarily on the technical aspects of disc development, some instructional and visual design considerations were addressed. Lists of technical standards, instructional formats, and aesthetic considerations are provided. Schwier criticizes the acceptance of a general IV design model and encourages creative design with respect to instructional and perceptual foundations. The importance of thinking instructively is stressed throughout the design process. Although IV is discussed mainly as a training tool, the author describes design considerations for many different types of learners. No attention is given to program evaluation or budget requirements. Schwier is involved in IV development at the University of Saskatchewan.

Design Component Interaction

Once an IV program has been produced, all of its components will operate within one interactive system. In order for these components to interact consistently within the system, highly significant component interactions must be identified and considered throughout the design process. Unfortunately, very little has been published on the processes involved in IV design. Some design components require a continuous updating process such as formative evaluation planning and budget management (Floyd & Floyd, 1982).
But most components have less obvious, but important, interrelationships such as behavioral objectives with flowcharting, and available audio and videodisc space with script length and storyboarding. Very little has been done to identify the processes by which these factors are interelated and controlled for during the overall IV design process.

Kearsley (1988) described how current software applications could serve as tools for managing specific components of IV design. Text, graphics, and data-processing programs all offer specific attributes that could assist designers in organizing, integrating, reproducing, and displaying IV design information.

Figure 2 is a summary of Kearsley's suggesested applications for software tools in the IV development process.

Analysis	Design	Development	Production	Evaluation
Needs Assessment Data collectioning and report generating. **Task Analysis** Sorting and printing task lists. **Behavioral Objectives** Providing alternatives for conditions, standards, and actions. Cross referencing with lessons and test items.	**Sylabus Development** Classifying and sorting the task hierarchy and behavioral objective information. **Storyboarding** Specifying screen design, animation sequences, and feedback sequences. Storyboard printing, and transfering data files.	**Writing** Word processing and editing scripts. **Art** Generating graphics and databasing slide, photo and video images. **Recording** Generating cue lists for production.	**Programming** Authoring **Printing** Publishing documentation and information. **Editing** Generating computerized editing lists.	**Field Testing** Presenting online formative questions during pilot-testing. **Cost/Benefit Analysis** Calculating specialized worksheets and spreadsheets.

Figure 2. Software tool applications

Although Kearsley describes the benefits of software applications that are aligned with specific design components, he emphasizes the need for software that integrates all of the various component's. Such software would facilitate the cross-referencing and tranfering of information from one design component to another. This type of integrated software tool could increase designer's productivity while providing them with system for monitoring component interaction.

Justification for a Comprehensive Model for Software Tool Integration

The 10 development models illustrate that different development philosophies and situations require different emphases. Yet, the selection and integration of software-based tools requires identification of all tasks within the IV development process which require documentation or the procesing of information. A comprehensive model must be able to: (a) embrace the full range of philosophies and (b) remain flexible and adaptable to these emphases. Each emphasis involves a different set of design tasks, and thus, would require a different sets of tools. In addition, individual work methods also dictate the types of tools that best fit development needs (Bork, 1989). For example, an individual who manages a large, in-house IV development team working on projects for external clients probably requires more sophisticated project management tools than a person who works by himself or herself on a small in-house project.

Based on the literature, it appears as though a comprehensive design model, identifying the tasks required within a variety of different areas of emphasis, would facilitate the overall IV design process and the identification and integration of specific software-based tools. In order to identify and integrate a comprehensive set of software-based tools that mcan be used within the development process, all possible design tasks must be identified.

CHAPTER III

DESIGN OF STUDY

The purpose of this study was to develop and evaluate a computer-based system for managing the design and pilot testing of interactive videodisc (IV) programs. In order to effectively define and integrate all aspects of systematic IV program development, the design of this study was in five major phases, (a) collection of models from the literature, (b) collection of models used by practitioners, (c) comprehensive model development, (d) software system development, and (e) software system evaluation. These phases provided information required to fulfil each of the objectives previously defined in Chapter I. The first phase, the literature model collection, consisted of a review of the literature pertaining to the design of IV programs and the extraction of design process models. In second phase, practitioners were interviewed and their design process models were either collected or constructed. A comprehensive design process model synthesizing both the literature and practitioner models was then created in phase three. The fourth phase consisted of the development of a computer-based design management system based upon the comprehensive model. Once the computer-based system was developed, the system was then evaluated in the fifth and final stage by the group of IV design professionals.

Phase 1- Collection of Models from the Literature

The review of literature focused on gathering all of the current literature related to the design of IV programs. This review concentrated on both the design process and its necessary components. Instructional design, visual design, program design, and project management concerns were addressed during the review. This review included a manual search of all related text and periodicals as well as a secondary computer search for further related publications. These searches were limited primarily to models and descriptions that had been published within the last five years to ensure that the information is pertinent to the current media environment.

After the related information was collected, key IV design components and processes were identified and summarized for each source. The key tasks and concerns were then categorized to assist in analysis and comparison. Once all of the pertinent sources had been summarized, the results were be analyzed for similarities and differences. Because each of the models focused on a particular approach to IV design, it was assumed that no one model would contain all the identified tasks and concerns.

Phase 2- Collection of Models Used by Practitioners

The primary focus of the second phase of the study was to identify and collect IV design process models that were being used by practitioners. The collection involved a series of taped interviews with four professional IV program designers.

Interview design. Prior to the interview, each expert was asked to collect any formal types of documentation that they used in the IV program design management process. They were asked to bring this information to the interview and to provide copies of any documents they would be willing to share.

Each interview consisted of both structured unstructured questions. The interviewer first described the purpose of the interview and asked the expert if he or she had any questions. Next, the expert was asked to describe his or her own IV design process, including both program design and project management tasks. The expert was also asked to identify the types of hardware and software tools/applications he or she used during the development process. If the expert provided any additional forms of process documentation, he or she was asked to describe the details and application of the document(s). The interviewer noted all pertinent information on pencil and paper and asked for elaboration when necessary.

Model collection. After each interview was completed, a design process model was developed from the interview information for any expert who did not provide documentation of his or her design process. Once a model was developed, it was then presented to the expert who was involved. The expert was asked to evaluate the model as being an accurate representation of his or her process. Further changes were made to the models when necessary.

After all of the interviews had been conducted, the four resulting adjusted models were summarized. The process depictions and component interrelationship information were also summarized and noted for integration during software development.

Phase 3- Comprehensive Model Development

In the third phase, a comprehensive model was developed that combined the project elements and tasks from the literature models in phase 1 with those collected in the practitioner interviews in phase 2.

The first step in this process was to analyze all of the identified tasks and to eliminate redundancies between the models. In order to eliminate all redundancy, a standard set of terminology had be identified since the same task may be referred to in a variety of different ways by the different sources. The models were then combined one by one and rechecked for redundancies and terminology consistency. Once all of the models had been combined into a comprehensive model, the included tasks were analyzed and roughly sorted according to their emphasis. Major project elements were then be identified and formally defined. The final step in the development of the comprehensive model required the categorization of the tasks under the major project elements.

Phase 4- Software System Development

Phase four consisted of the development of a computer-based IV design management system from the new design model and the design process information summary. This system consisted of software tools that allowed for the logical integration and control of all interrelated design elements and tasks. IV project elements and tasks were incorporated within specialized design management and development tools. Design management tools were developed that gave IV designers the ability to previsualize designs, document and sort program design information, print specialized reports, and pilot test a programs content and structure. Additional development tools were developed to integrate and assist in the tasks identified in the comprehensive model.

The IV design system was developed using Silicon Beach's SuperCard authoring system and a Macintosh SE microcomputer. Extensive printed documentation was developed to instruct users in the most effective way to integrate and operate the system.

Phase 5- Software Evaluation

Once the computer-based design system was completed and debugged, the IV design management software was evaluated. Three IV design experts were asked to examine and use the design system and to evaluate its utility. The design experts used in this phase of the study were different from those used in phase two.

Software evaluation design. The design of the software evaluation procedure was as follows:

Meetings were arranged with three IV design experts. During each meeting, the researcher explained the purpose of the study and the software system. The researcher then demonstrated each of the software tools and familiarized the designer with the user interface. During this time, the expert was encouraged to ask questions or make preliminary comments. The experts were asked to spend a minimum of two hours pilot testing the system outside of the initial meeting. During this time they were asked to begin designing a small section of a program of their choice and explore each of the tools the software system provides. Each expert was given a set of formative and summative evaluation forms to record their reactions, comments, and evaluation's during the pilot-test period.

Once all the meetings, pilot tests, and evaluations had been completed, the evaluation forms were collected and analyzed. Response similarities and differences were noted. The software system and documentation were then revised, according to suggested changes whereever it was necessary and possible.

Experts

The experts for this study were seven professional IV program designers. Each of these designers had previously worked on the design and development of at least two IV programs. These designers were selected from seven different public and private IV development organizations.

Reimbursement

In order to encourage IV design professional participation, each participant was offered an hourly consulting fee and received one personal copy of the completed software and documentation. Each participant also could request one complementary copy of the completed study.

Information Collection

Each of the phase two interviews were recorded using a portable audio tape recorder. The recorder was placed in a nonobtrusive location and random pencil-and-paper notes were taken for use by the interviewer and to reduce the intimidation of the designer.

During the software evaluation phase, the researcher took paper-and-pencil notes related to comments and questions made during the initial meeting. Expert evaluation data and pilot-testing comments were recorded using a series of printed forms.

Facilities and Equipment

The only facility needed to conduct this study was an office. This facility was provided by the Training and Development Research Center at the University of Minnesota. Funding for expert consulting fees was also provided by the Training and Development Research Center at the University of Minnesota.

A Macintosh SE microcomputer and printer was required to author the design system software and transcribe the interview information. The computer was also loaned to participating designers who did not have access to a Macintosh computer during the design system evaluation phase. A Macintosh SE computer was made available through the University of Minnesota's and Apple Computer's Project Minnemac for these purposes.

CHAPTER IV

RESULTS OF THE STUDY

The computer-based system for managing the design and pilot testing of interactive videodisc programs was developed and tested according to the six phases described in Chapter 3. The literature and practitioner model collection phases concluded with the development of a comprehensive interactive videodisc (IV) design model. A software system, the Interactive Design Management System (IDMS), was then constructed to facilitate the integration of the project elements and tasks identified within the model. Once the software system was completed it was then evaluated by three IV program design experts. The following sections describe the information that resulted from each phase.

Phase 1- Collection of Models from the Literature

The results of the collection of models from the literature is contained in Chapter 2. During the review of literature 10 different IV design process models were identified. As discussed in Chapter 2, the scope and focus of each of these models varied according to the author's backgrounds. Because the terminology and organization varied between each model, five categories were identified to assist in organizing and comparing the information. Figure 1 in Chapter 2 contains a matrix that summarizes the tasks described by each of the literature models under the general categories of front-end analysis, content outline, program design, production requirements, and project management. Most of these models emphasized the instructional design tasks involved in IV program design. Less emphasis was given to the tasks and specifications required to manage and maintain the instructional design during the development of the program. Although all of the authors referred to possible interactions between the design tasks, DeBloois's (1982) original model was the only model that depicted the possible systematic interchange and/or sharing of information between tasks. Most of the authors also referred the use of computer programs and paper-based forms for managing various components of the design process. However, Kearsley (1988), in an article separate from his model, was the only author to identify specific applications for software tools within the IV design process (see Figure 2).

Phase 2- Collection of Models Used by Practitioners

In the second phase of the project a series of interviews were conducted with IV design experts. The purpose of these interviews was to identify if practitioners were using design models and if so, to collect the models and information on any other tools or software they may use. Eleven IV program design experts were identified through the Minnesota chapter of the International Interactive Communications Society and through the recommendations of other IV practitioners.

A set of structured and unstructured interview questions was then derived from the information gathered in the collection of models from the literature (see Appendix A). The initial questions focused on identifying the key stages and tasks required in IV program development. Latter questions addressed the tools and techniques the practitioners used to manage and transfer the different types of design-related information. In addition to the questions, a sheet containing a matrix summarizing the tasks identified in the literature models was also constructed (see Appendix B). The tasks in this model were organized under the same set of categories as the individual literature models. This model was intended to serve as a possible organizer during the interviews.

Interviews were then conducted with IV design experts from seven organizations, including Arthur Andersen Consulting Inc. the Center for the Advancement of Learning Technologies of the University Minnesota - Duluth, Emerging Technology Associates, Minnesota Extension Service, Northwest Aerospace Training Corporation, and Wilson Learning. During these interviews, the practitioners were asked a series of questions related to the tasks involved in IV program design, project management techniques, and tools used in the design process. When asked to relate their responses to the summary of the literature models, four of the practitioners produced or described detailed models that were unique to their organizations. Due to the proprietary nature of this information, source names will not be associated with the individual models; therefore, they are labeled models A, B, C, and D.

After all the interviews were completed, only the information collected from the four practitioners using formal IV design models was analyzed. The information collected from the other three practitioners was dropped for lack of detail or organization. Because each of the practitioner models was in a different form (see Appendix C), the models were summarized to facilitate analysis. The following sections highlight the specific characteristics of both the origins and emphasis of each of the models. These sections are followed by a overall summary, comparing all four of the models.

Model A. Figure 3 summarizes a model used by an IV group within a large organization that produces educational and training programs for both

Model A

Development
- Identify topic
- Identify audience
- Assess available budget
- Define broad objectives
- Develop concept flowchart
- Identify SMEs
- Project management schedule
- Project summary report
- Marketing department preview

Design-Script
- Def. measurable objectives
- Define levels of interactivity
- Est. finished disc length
- Construct test items
- Select SMEs
- Extract SME information
- Edit SME information
- Produce content outline
- Create model flowchart
- First draft of script
- Dev. visual script
- Team rev. of nar. & vis. script
- Revise script doc.
- Produce final script
- Final detailed flowchart

Design- Print matter
- Identify addit. instruct mat.
- Extract info. from SMEs
- Prod. student & admin mater.
- Proofread materails
- Corrections
- Arrange printing

Preproduction
- ID char. & loc. from script
- Gen. new script with info
- Create shot list
- ID studio and loc. req.
- Secure prod. crew & house
- Identify talent needs
- Est. prod. values w/team
- Design set and floor plan
- Produce prompter script
- Secure talent
- Scout & secure locations
- Create prod. schedule
- Collate all prod. docs.
- Create production report

Graphic Production
- ID standard graphic formats
- ID production houses
- ID txt for menus & txt frames
- Design animat. seq.
- Create animat. storyboards
- Prod. menus and still frames
- Revise animat. & still frames
- Create window burn
- Review tape for corrections
- Revise window burn tape
- Create submaster.

Production
- Prod. crew meeting
- Rehearsals
- Shoot loc & studio
- Window burn tapes
- Produce audio
  - Narration
  - Music
  - Transition effects

Postproduction
- Produce EDL from work tapes
- Design disc architecture
- Collate all postprod. data in report
- Off-line edit all video motion seq.
- Review text & anim. sequences
- Create video transitions
- Familiarize editor w/VD specs.
- Online editing
- Produce actual EDL
- Review window dub
- Revise edit master
- Submit mastering form
- Check disk

Programming
- Code routines
- Log check disc
- Log frame #s into flowchart
- Check logic with disc
- Revise programming
  - Communication tracking sheet

Replication
- Replicate floppy disks
- Replicate videodiscs
- Replicate documentation

Evaluation
- In-house pilot-test
- SME pilot-test/review
- Beta test on site
- General distribution

Figure 3. Model A

internal and external applications. At the time of the interview, the model had not been formally documented, although the practitioner was able to describe the IV design process and tasks in great detail. In order to construct a formal model, the interviewer constructed a representation of the described model after the initial interview and then had the practitioner evaluate the representation prior to a later meeting. The tasks defined in this model are primarily the concerns of a staff member who manages IV program development.

The primary emphasis of the model is on production-related design concerns. Very little preliminary analysis is depicted within the model because other departments within the organization typically define the problem and general objectives before presenting the project to the IV group. Since most external applications are intended for the public sector, the development of noncommercial marketing materials is also considered to be a part of the design process. During the interview, the practitioner emphasized the need for constant communication among the members of the development team. However, the model contains very few tasks or procedures for facilitating such interaction. Because the organization is not market-driven, very little emphasis is placed on any form of evaluation. The only specialized tools used in the design process are a series of worksheets for estimating costs and documenting the reviews and revisions. A word-processing program is used to manage all design documentation.

Model B. Figure 4 summarizes a formally documented model used by a computer-based instruction department within a large organization that produces training programs for private and public sector clientele. The original documentation of this model contained a list of the main

Model B

Planning Phase

Organize project team
Confirm project approach
Estab. time control, reporting, QA procedures
(All processes below are continuous)
Detailed planning
  - Finalizing detailed work programs
  - Issuing individual work schedules
  - Setting up time control
  - Prep. progress reporting stdrds.
  - Prep. qual. assur. schedules
  - Prep. work paper doc. stdrds
  - Orient project team
Time estimation and monitoring
Project status reports
  - Work completed to date
  - Work in progress
  - Work not started
  - Project issues and recomm.
    - Sig. content & design issues
    - Schedule problems
    - Communication issues
    - Action Plans
  - Critical date schedule
  - Resources & material requir.
Internal and external QC &QA

Needs Analysis Phase

Confirming training plans
  - Rationale
  - Course goals
  - Content scope
  - Audience profiles
Audience needs
  - Primary audience
  - Secondary audience
  - Tertiary audience
Develop objectives
  - Behav. objectives
  - ID logical skills
  - Enabling objectives
Choose authoring sys.
Rev. author sys. param.
Define course content
  - Review existing mat.
  - Gen. orig. content drafts
Analyze content
  - Content relevancy
  - Content emphasis
  - Level of detail
  - Level of application
  - Testing for objectives
Planning course architect.
  - Menu structure
  - Instructional modules
  - Alternative learning tracks
Define & sequence content
  - Detailed content outline

Design Phase

Finalize Design
Define product image
  - Video consistiency
  - CG consistency
  - Standard icons
  - Screen design
  - Text design
  - Packaging
Determine validity/realism
Define interactivity
  - Overall flow
  - Module maps
  - Topic sequencing
  - Points of interaction
  - User control strat.
Video treatment
Media/objective alloc.
  - Motion video
  - Still frame
  - Computer graphics
  - Audio
  - Printed support
  - Est. storage space
  - Number of each
  - Est. time
Detailed design document
Design walkthrough

Development Phase

Create scripts and boards
  - Motion video
    - Loc descript.
    - Set descript
    - Creative treatment
    - Shot list
  - Audio Usage
    - Narrative script
    - Sample music
    - VO script
  - Still images
    - Brief description
  - Graphic Images
    - Brief description
  - Computer output
    - Screen design Stndrds.
    - Text standards
    - Graphics standards
    - Answer judging
    - Feedback
    - Touch areas
    - Audio
Video prod. planning
  - Video budget
  - Video locations
  - Video sets
  - Types of actors
  - Actor contracts
  - Prop list
  - Production schedule
  - Postprod. issues

Development (cont.)

Create video database
  - Number of shots
  - Type of shots
  - Shot loc.
  - Actor involvement
  - Wardrobe needs
  - Prop needs
Draft of print mat.
  - Student workbook
    - Getting started
    - Course overview
    - Objectives
    - Agenda
    - Course map
    - Module overview
    - Reference mat.
    - Key illustrations
    - Glossary
  - Job aids

Figure 4. Model B

Video Production Phase

- Video treatment
- Video scripts
- Eval. shoot locations
- Talent search
  - Auditions
  - Contracts
- Video shooting
  - Studio shots
  - Location shots
- Recording Audio
  - Voice-over narration
- Videotape editing
  - Disc mapping
  - DVE
  - Edit decision lists
  - Check disc production

Authoring Phase

- Training on authoring system
- Author course segments
- Review course segements
- Development testing
  - Content sequence
  - Content flow
  - System use
  - Points of interaction
  - Reference material
- Videodisc/Software integration
  - Program with check disc
- Team testing of integrated program
- Courseware revision
- Draft program maintenance manual
  - Write program assumptions
  - Write program defaults & stndrds.

Validation Phase

- Create eval. questionaire
  - Anal. content accuracy
  - Anal. achieve of object.
  - Anal. course flow & log.
  - Anal. course timing
- Target aud. validation interviews
- Create pre & post tests
- Conduct formal evaluation
- Document validation results
- Revise course materials
- Finalize document
- Software maintenace

Figure 4. Model B (continued)

components of the development process followed by a detailed description of each of the tasks involved in each component. This model is referenced by the project manager and staff, and it is also used to familiarize clientele with the organization's development process.

This model addresses all aspects of the design process quite equally. However, because most IV programs are produced for external clients, there is an extra emphasis on time scheduling, reporting, and quality assurance. In order to ensure that the product is completed on time, a variety of project and personnel management tasks have been incorporated. Design documents and reports are generated thoughout the design process to ensure that the design and perceived product will fit the client's needs. This documentation and a system of formative internal and external checks and reviews also serve as methods for ensuring the quality of the end product. The practitioner described many software applications that are used during the design process. These applications included project management programs for monitoring project schedules, spreadsheets for monitoring costs, data bases for managing video production data, and authoring systems for piloting designs.

Model C. Figure 5 summarizes a formally documented two-part model used by an internal IV development group that specializes in technical training. The development model provides a general overview of the major components of the development process. Within design, preproduction, and production components of this model occur the detailed tasks described in the production model. The tasks within each component of the production model were originally depicted on the vertical axis of a series of two axis matrixes. The horizontal axis of these matrixes contained a time line for denoting the expected initiation and completion dates for each task.

Model C

Development Concerns

Organization

- Needs analysis
- Gather data
- Review existing doc.
- Determine return/investment
- Determine available resources
- Develop outline & treatment
  - Task analysis
  - Goals and objectives
- Develop project definition
- Create design document
- Assemble project team
  - Project manager
  - Subject matter experts
  - IS/IV designer
  - Video/Print writer
  - Video producer/director
  - Programmer
- Obtain necessary review & approval

Master and Release

- Master and replicate
- Create marketing doc.
- Order packaging
- Diskette reproduction
- Maintenance and revision

Design

- Dev. instruct. flowchart
- Complete video scripting
- Develop storyboards
- Develop graphic design
- Prepare development doc.
- Plan/prepare prod. schedule
- Obtain review and approval

Preproduction

- Studio and loc. arrangements
- Select talent
- Arrange set and props
- Review existing visual materials
- Shoot still-photography
- Obtain review and approval

Production

- Studio and loc. shooting
- Graphic production
- Edit pre-master tape
- Computer programming
- Review check disc
- Obtain review and approval

Validation

- Program testing
  - programming
  - reliability
  - usability
- Perform video revisions
- CG and text revisions
- Revise premaster tape
- Obtain review & approval

Figure 5. Model C

Production Concerns

Design

Task analysis
Objectives and goals
Scope and sequence
Concept flowchart
Instructional Strategies
Lesson design
How to assess stud. perf.
Advisory review
Design summary report

Script

Content outline
1st draft of script
Team review
Final script
Detail flowchart

Graphics

Dev. style/format
Storyboards drawn
CG text and menu frames
Graphic design review
Animation audio recording
Audio to video transfer
Pre-prod animation
Animation production
Transition graphic prod.
Still frame graphic prod.
Transition review
1st still frame review
1st animation review
Still correct / revise 1
Anim. correct / revise 1
2nd still Review
2nd animation review
Correction / rervision #2
Create submaster
Review submaster
Correct / revision #3
Edit changes to submaster
Completed submaster

Print

Select Doc. type
Design format
Write
Review
Revise
Field test
Graphic design
Typeset
Keyline
Read galley proofs
Revise
Final review
Send to printer

Studio

Production design
- Design environ.
- Affect on viewer
- Prod. values

Set design
Set construction
Props
Set-up and staging
Talent selection
Script time estimate
Shot list
Prompter scripts
Shoot
Window-burn work tapes

Location

Talent Selection
Site Selection
Shoot
Win.-burn work tapes
Obtain library footage

Edit

Log source tapes
Disc architecture
Create EDL
Timecode calc.
Primary edit
Review edits
Clean up edits

Programming

Review script & flowchart
Code spec. routines
Code logic
Log check disc
Add frame numbers
Logic test
Beta test
Team review
Revise
Replicate

Figure 5. Model C (continued)

This model was designed to be used and referenced by the entire development group.

Although most programs produced by this group are highly technical in nature, the models provide very little detail related analysis or instructional design strategies. The model places much more emphasis on production-related details. Similar to the role of the group using Model A, the group using Model C was generally assigned to projects with the programs subject and objectives already defined. One major difference however, is that this group most often researched that the return on investment will be substantial enough to justify the development costs. Even after the approval of a project and its initiation, a number of approvals must be secured throughout the development process. The practitioner described the use of a spreadsheet application for designing videodisc architecture and calculating disc space. Word-processing software was used to create most development documentation.

Model D. Figure 6 summarizes the most sophisticated of all the practitioner models. This detailed model is used by an interactive videodisc production department within a large organization that produces training programs for the private sector. The original model was organized as a map of milestones. Separate documents described each of the tasks that fell under each of the milestones. This model was designed to be referenced by the project manager and in some cases, to be used to describe the development process to external clientele.

Because this organizations IV products are focused on the private sector, this model emphasizes client or market satisfaction and quality control. The design process is broken down so that a predesign concept can be fullly

Model D

Phase 1
Identify Solution

Plan integrated offering
- Def. training comp. & desired results
- ID target aud. profile
- Def. purpose of each solution & desired outcome
- Def. program user prereq.
- Determine seq. of learning events
- Survey market
- Format (hardware) preferences

Dev. plan to obtain design funding
- Estimate cost of design

Management eval. & sets priorities steps
- Approve relevent budgets

Phase 2
PreDesign

Plan for design
- ID resources/designers
- Recruit and contract resources
- Brief resources
- Project team meeting
- Agree on maintenance of sytem files sharing structures and create doc.
- Dev. and doc. schedule

Dev. conceptual design frameworks
- Identify content
- Hard/software delivery consid.
- Doc. learning and perf. obj.
- Dev. testable design flow
- Test with int. aud., eval & refine
- Rev. team input for design doc.

Produce design document
- Def. scope of training
- Desc. of target aud.
- Task analysis
- Def. measure and eval. methods
- Learning obj. & inst. design
- Activities and desired outcomes
- Delivery sequence
- Video req. & senerios
- Graphics design req.
- User interface
- Deliverables
- Tailoring plan
- Marketing and support req.
- Hard/software requirements
- Print collateral materials
- Testing strategiews

Phase 2
PreDesign

Internal quality review and revision

Develop plan for obtaining approv.
- Budget req.
- Est. resource, budget & time req.
- Marketing bus. plan
    - marketing strategy
    - pricing & est. return on invest.
    - prospect targ. aud.
- Prep & pres. plan to top manag,

App. bus. plan, & budget

Dev. & set respon for qual. stand.

Trademark unique component

Rev. team app. predesign doc.

Phase 3
Design Document

Develop
- Design principles
    - Instructional design
    - Engine outline
    - "Treatment" flowchart
        - Content orient. for writing
        - Visual orient for graphics
        - evaluate projects current structures
        - design deliv. envir: menu, install, facilitor utilities and documentation
    - Major content assumptions
        - learning points to cover
    - Major product assumptions

Specifics:

Standards/Specifications document
- Disc real estate
- Video art
- Video text
- Rtg art
- Rtg text
- Transitions
- Production values
- Visual design /c olor pallette
- Print collateral
- Product & module titles
- Hardware & mem requir.
- Hardware diagrams
- Page numbering system
- Software syntax for branching
- Change/ update approval
- Change/ update informing system
- license aggreement

Define writers specifications

Figure 6. Model D

Phase 3(cont.)
Design Document

Production
Writing
- Write treatments
  - Official product description
  - Edit product descrip.
  - Outline of learning points
  - Edit ouline
  - Descrpt. of content present.
  - Branching instructions

Project Direction
Refine budgets and timelines
Get team and codeveloper signoff

Phase 4.1
Design & Mkting Strategy

Writing
- Draft 1 scripts to standards & spec.
  - Electronic script page
  - Audio/video dialogue w/rough time
  - Text
  - DVE descript.
  - Rtg directions
  - PC gen. sound directions
  - Stage directions
  - Loc. descriptions
  - Character descrioptions
  - Fill in draft fields: mod, date, text, pg# same as, character, loc, type.

Team review
Overall project structure based on draft 1
- Document sw related notes
- Design data flow: (eg. bookmarks, records)
- Finalize engine/node decomposition
- Design data file struc. & ID compat. issue
- Design new hardware-related structures, code overlays, file systems.
- Eval. sys ampat. & ID needs for new coding file creation, editing and doc,
- Review software logic

Marketing
- Execute marketing strategy
  - Dev. mkting materials, fact sheet
  - Edit fact sheet
  - Announce prodct to sales and est. date
  - Itemize product components for invent.
  - Estab. unit costs for comp.
  - Estab. hardware config.
  - Initial bill of mat. & invent #s/

Project direction
- Dev. "production" flowchart
  - define structures
  - Provide param. for writing
  - Test struct. & highlight incon.

Phase 4.1 (cont.)
Design & Mkting Strategy

Writing
- Write draft #2 scripts
  - Electronic script page
  - Aud/video w/ real timing
  - VO audio, A&B w/ real timing
  - Text
  - DVE use as per diesign document
  - RTG use as per design document
  - PC gen. sound direction consist.
  - Stage direction consistency
  - Loc. descript consistency
  - Character description consistency
  - Draft field consist: mod, date, disc, bkg, text, DVE, fmt, pmt, pg#, script, same as, char, loc.

Team review darft #2
Software follows up draft #2
Project direction
- Modify "production" flowchart
  - Define structures
  - Cement global calls
- Update timelines and budgets
  - Justify timeline & budget changes or modify scripts to fit initial plan

Writing
- Continue draft review stage

Editorial review of final script
Team review of final draft
Graphics
- Design print collateral materials
- Design video art, video text, RTG art & text

Phase 4.1 (cont.)
Design & Mkting Strategy

SW Eng
- Finaliize engine design
- Finalize nodal design
- Finalize hardware drivers design
- Finalize data files structures design
- ID req. mod. to current dev. tools and design and document alterations
- Create simulation of the program and test engine and design

Cust. support/marketing
- Alpha test
- Rewrite script based on Alpha feedbk.
- Rev. team sign-off & commit to design
- Product components stock
  - Number of laser discs
  - Number of floppy discs
  - Nu mber of packiaging materials

Quality rev. of design of mat. before prod.

Figure 6. Model D (continued)

Phase 4.2
Pre Production

Design
- Full review by specialists

Audio/Video
- Review by media team
- Break out script
- Set and lighting design

Editorial
- Copyediting
    - Content continuity
    - Syntax
    - grammar check
    - spelling and proof
- Integrated feedback from rev.
- Proof flowcharts w/ "prod" flowhart and group
- Standards/Specifications doc.

Software
- Review ne changes to software syntax

Graphics
- Proof graphics fields
- Complete script breakout
- Review set and lighting design
- Review print collateral vendors

Phase 4.3
Production

Audio/Video
- Complete production
    - Create continuity logs
    - Create video and film logs
    - record in Production Master Script
        - Script changes
        - Actual timings
        - Best takes
        - Source reel numbers
        - Actual talent data
- Produce VO audio

Graphics
- Video art and logs
- Proof video text
- Vidifont text and logs
- Proof vidifont text
- RTG art
- RTG text
- Proof RTG text
- Produce print collateral (except content of guides and workbooks).

Premaster
- Duplicate audio and video masters
- Edit off-line
- Build DVEs
- Dupe VO audio masters
- Edit VO audio
- Transfer audio to 1" tape
- Assemble source reels
    - Video art
    - Video text
    - Still frame build

Phase 4.3(cont.)
Production

Project direction
- Input sequence map
    - Sort
    - Check
- Create/define music
    - Original
    - Library
- Complete "assembly" flowchart
- doc. actual prod. addit. & changes

Software engineers
- Assign coding tasks
- Input data to CG
- Code engines & modules & doc.
- Code hardware drivers

Writers/Editiorial
- Collateral mat. composed, editied
    - Workbooks
    - Coodinators guide
    - Salespersons guides
    - implementation handbook

Phase 4.3(cont)
Production

Graphics
- Design collateral materials

Review team sign off

Project Direction
- Check source reels

Premaster
- Revision
- Edit on-line
    - Edit numbers
    - SMPTE time code
    - editior comments
    - master reel numbers

Sweeten master tape
- Source materials
- Master 1" tape and review
- Revisions to 1" tape
- Duplicate 1" master tape
- Proof disc and press replicate

Software engineer
- Finish coding

Project direction
- Create final flowchart
- Review disc pronhlems
- Create disc map log

Project director signs off

Figure 6. (continued)

Phase 4.4
Post Product Production

Software engineer
  - Complete all frame related files
  - Integrate system and test

QC test/recommend/recycle
  - Software
  - Collateral
  - Design

Graphics
  - Sched. prod. of print collateral
  - Replicate collateral after QC
  - Proof quality of print materails

Project Direction
  - Authorize rev. until complete

Sign-off review team

Phase 5
Beta Test and Evaluation

Register product

Final product check
  - Legal criteria met
  - Collateral mat. complete
  - Item and hardware #s in place
  - Markting info avail.

Replicate & assemble for field test
  - Discs, software, collateral

Design beta test & ID test sites

Ship and client field test
  - Rework based on field feedback
  - Dev. support procedure
  - Release final product
  - Mass replicate & stock
  - Sell
  - Ship orders

Phase 6
Ports, Product Releases, and Archiving

Ports
  - Schedule secondary ports after beta

Product releases
  - Schedule different hardware config.

Archive
  - "Originals" storage
    - Paper script/logs
    - Electronic script logs
  - Premaster
    - Orig. glaas master
    - Orig source reels
    - Orig 1" master

Software engineer
  - Floppy/back up tape storage

Review team session on what occured

Phase 7
Product Maintenance

Database of bug and feedback info

Product updates or re-releases

Testing portability to new systems

Testing life of components (soft & hardware)

Figure 6. Model D (continued)

described and reviewed prior to the onset of any technical specification. Once the predesign concept has been authorized, the logistics of the production of the program are developed. Progress reports and reviews are used throughout the design process in order to obtain client or marketing department authorization and to ensure that the objectives are being addressed. Since this organization produces specialized programs, most software is custom-written to provide advanced levels of interactivity and recordkeeping. The comprehensive documentation required to facilitate this programing is evident throughout the design model. In addition, design consideration is also given to facilitate future software compatibility and transfers "ports" to other hardware systems. The practitioner described the use of data base and filing software to manage the large amounts of complex data that are required in the program development process.

Model Comparison. Comparison of the collected practitioner models shows there are significant differences in detail and emphasis between each of the models. These differences appear to be related to four factors: (a) the audience the model is designed to serve, (b) the organizational positioning and size of the design team, (c) the type of clients served by the organization, and (d) the use of internal and external production groups.

Because all of the models are designed to be used as a reference and planning tool, the primary audience is the person(s) responsible for managing the IV program development. However, the tasks outlined in the production section of Model C were designed to be referenced by the entire IV group in the form of individual task time lines.

Models B and D's audiences are even broader because they are referenced by the IV staff and also used to familiarize external clientele with the organization's IV program development process. Because of this external application, these models tend to include many tasks that are related to quality assurance.

Organizational positioning and size of the design team is evidenced within the scope of each model. Models A and C show limited attention to task involved in front-end analysis. The IV design teams using these are positioned within organizations that typically define the goal and objectives of IV programs prior to the team's involvement with the project. Therefore, the teams focus primarily on the production-related design tasks required to transform the predefined idea into a product. In addition, it was also quite evident from all the interviews that smaller, less prolific IV development teams are less likely to have a formalized model because there is less need to communicate and standardize the process. This is also shown in Model A.

Similar to the issue of audience, the models also vary according the type of clients the organization serves. Models B and D are representative of organizations serving external, private sector clientele. The models are very highly detailed from conceptualization to completion to ensure quality and client satisfaction. This high degree of control may be considered too time-consuming for organizations thast have less or no market dependence.

An IV development team's use of internal or external production groups might effect a models emphasis on production-related details. However, since all of the teams had external groups perform most of the production-related tasks, there is little difference in this attribute between models. Only Model D, with its detailed programming requirements shows a significant emphasis on any type of technical task detailing.

After considering these factors, it was quite evident that the overall scope and emphasis of each team's model varies according to the organization to which they are ultimately accountable. Organizations supported by external clientele ensure accountability through a sophisticated system of checks and balances. They differ from internally focused teams by placing less emphasis on accountability, possibly because of internal constraints on the design process.

Phase 3- Comprehensive Model Development

Once the literature and practitioner models were collected and summarized, a comprehensive Interactive Design Management System model was created. This model was formed by summarizing, categorizing, and editing the tasks identified in the collected models. The following sections briefly describe the processes that took place during the creation of this model.

Model summary. The summary process began with the tasks defined in the literature model summary matrix developed for use in the Practitioner Model Collection phase (see Appendix B). The tasks from this matrix were placed in one list, and an attempt was made to order them chronologically. Since many of the tasks may be reoccurring or they may occur simultaneously during program development, an exact chronology was not possible. The tasks identified within each of the practitioners models were then extracted and ordered using the same process. Each of the practitioner models was then compared, task by task, to the literature task list. Tasks in the practitioner lists that were redundant with the literature list were eliminated. Next, all remaining tasks that were unique to the practitioner model were chronologically inserted within the literature list to form a new combined task list. This process was repeated with each of the remaining practitioner models using the newly combined list for comparison. The result of this process was a chronological list of all unique tasks identified in either the literature models or within the practitioner models.

Task categorization. The combined task list was then analyzed and major IV program development elements were identified. Eleven elements were defined by organizing the tasks in the combined task list into developmental stages. The major categories defined in the original literature models and practitioner models were used in arriving at elements names. Each of the elements was then formally defined to facilitate task categorization (see Figure 7). Next, the tasks were placed under their corresponding elements.

Final model editing. Since the combined model contained a number of detailed and/or specialized tasks that were specific to individual practitioner's needs, it was necessary to edit and revise the model into a more generally applicable form. Many of the practitioners models referred to tasks involving organization-specific forms or procedures. Once identified, these tasks were either deleted or, if deemed to be of general utility, rewritten in a more generic form. All tasks that focused on program marketing were eliminated since these tasks vary between organizations. The final step in this phase was to review the entire IDMS model to ensure that all of the tasks were represented at approximately the same level of detail. All identified inconsistencies were either rewritten or edited out. Figure 8 depicts the resulting IDMS model consisting of 191 tasks that are organized within 11 major design process elements.

Interactive Design Management System Elements and Definitions

I. **Project Management:**
Organizing, monitoring, and controlling all project related tasks and resources.

II. **Content Analysis:**
Specifying a programs conceptual and physical parameters based upon environmental needs and internal constraints.

III. **Delivery System Design:**
Specifying the configuration and capabilities of the authoring environment and delivery hardware.

IV. **Content Development:**
Identifying, collecting, organizing, and refining information into a workable instructional design.

V. **Definition of Standards:**
Constructing an overall set of program design rules and parameters to which all program content must adhere.

VI. **Preproduction:**
Identifying and controlling all production components prior to the production process. Generating all documentation necessary to accurately communicate the instructional design and production requirements to the media producers and talent.

VII. **Electronic Media Production:**
Creating and recording all necessary audio and visual materials.

VIII. **Postproduction:**
Organizing and integrating all recorded materials into a program according to a preconceived design.

IX. **Program Authoring:**
Developing a software-based program which controls an instructional programs delivery and interaction.

X. **Print Production:**
Identifying, designing, and producing all printed information.

XI. **Evaluation Design:**
Constructing strategies and instruments for validating a program.

Figure 7. IDMS elements and definitions

Interactive Design Management System Model

A. PROJECT MANAGEMENT

1. Define staffing requirements
2. Identify project team
3. Project team meeting
4. Establish standard reporting procedures
5. Establish standard method for documenting program updating and changes
6. Establish system for change authorization
7. Define critical dates and deadlines
8. Develop work status estimating, reporting, and monitoring procedure
9. Develop project work schedule
10. Develop formative quality assurance procedures
11. Develop individual work schedules
12. Develop integrated team work schedules
13. Develop file sharing and organization procedures
14. Establish problem reporting and resolution procedures
15. Estimate resources and materials required
16. Perceived cost/benefit analysis
17. Assess available budget

B. CONTENT ANALYSIS

1. Identify program topic
2. Conduct needs assessment
3. Define rationale for program
4. Define program goals
5. Define program scope
6. Identify primary, secondary audiences
7. Define audience characteristics
8. Define learning environment
9. Define program user prerequisites
10. Define content assumptions
11. Determine internal evaluation criteria
12. Determine external evaluation criteria
13. Develop behavioral objectives
14. Develop measurable objectives
15. Define instructional strategies
16. Define methods of measurement and evaluation
17. Define number of program copies to be produced
18. Produce project summary report

Figure 8. Interactive Design Management System model

Interactive Design Management System Model

C. DELIVERY SYSTEM DESIGN
1. Define hardware configuration
2. Choose authoring system or language
3. Define authoring capabilities
4. Define user interface(s)

D. DEFINITION OF STANDARDS
1. Define program defaults
2. Define intended affect on viewer
3. Define programs personality
4. Define feeling of programs environment
5. Define production values
6. Define video consistency standards
7. Define computer graphic design consistency standards
8. Define transition consistency standards
9. Define screen design consistency standards
10. Define menu usage consistency standards
11. Define icon usage consistency standards
12. Define color coding and usage consistency standards
13. Define narrative consistency standards
14. Define video sound effect consistency standards
15. Define video soundtrack usage consistency standards
16. Define video text consistency standards
17. Define print material consistency standards
18. Define computer generated audio consistency standards
19. Define computer generated text consistency standards
20. Define touch screen areas
21. Define types of user interactivity
22. Define input device usage consistency
23. Define learner tracking strategy

E. CONTENT DEVELOPMENT
1. Identify subject matter experts
2. Select subject matter experts
3. Conduct subject matter analysis
4. Interview subject matter experts
5. Conduct task analysis
6. Generate rough content outline
7. Review existing content materials
8. Write new content components
9. Modulize content components

Figure 8. Interactive Design Management System model (continued)

Interactive Design Management System Model

E. CONTENT DEVELOPMENT (continued)
10. Identify points of formative user evaluation
11. Pair evaluation points with objectives
12. Develop concept flowchart
13. Define branching alternatives
14. Define detailed content outline
15. Identify types of print materials
16. Outline print materials
17. Write first draft of narrative script
18. Define inner module interactivity
19. Select specific mediums for each program segment
20. Create detailed content flowchart
21. Pilot test content flow
22. Devise user introduction to system operation
23. Team evaluation of design
24. Revise design
25. Acceptance of final design
26. Create design document

F. PREPRODUCTION
1. Estimate video production budget
2. Identify existing footage & sources
3. Create video shot list
4. Define video treatment
5. Write location descriptions
6. Write set descriptions
7. Identify props needed
8. Write still frame descriptions
9. Create still frame sketches
10. Write descriptions of graphics
11. Write video effects descriptions
12. Write text frame descriptions
13. Write menu descriptions
14. Write animation descriptions
15. Create animation storyboards
16. Write stage directions
17. Identify studio requirements
18. Evaluate shot locations
19. Collect production bids
20. Secure prod. house and crew
21. Make location & studio arrangements

Figure 8. Interactive Design Management System model (continued)

Interactive Design Management System Model

F. PREPRODUCTION (continued)

22. Generate final narrative script
23. Create teleprompter script
24. Create voice over script
25. Design disc audio track usage
26. Estimate completed disc(s) length
27. Identify necessary talent
28. Audition talent
29. Select talent
30. Identify soundtrack styles
31. Identify sources of soundtrack mat.
32. Team review of script and storyboards
33. Create production schedules
34. Create production reports
35. Rehearsal

G. ELECTRONIC MEDIA PRODUCTION

1. Production crew meeting
2. Shoot studio video sequences
3. Shoot location video sequences
4. Produce still frames
5. Record studio audio
6. Record original music
7. Obtain "needle drop" music
8. Produce computer graphics
9. Produce animation sequences
10. Produce video text/menu frames
11. Review with design team
12. Create postproduction document

H. POSTPRODUCTION

1. Create window burn dubs of all source reels
2. Log window burn tapes
3. Create a disc map
4. Create rough edit decision list
5. Off-line edit rough edit tape
6. Review off-line edit tape
7. Revise edit decision list
8. Recalculate disc space usage
9. On-line edit
10. Create dubs of premaster reels

Figure 8. Interactive Design Management System model (continued)

Interactive Design Management System Model

H. POSTPRODUCTION (continued)
11. Review premaster
12. Revise and review premaster reel if needed
13. Print final edit decision list
14. Produce check disc
15. Review check disc
16. Revise premaster if necessary
17. Log check disc frame numbers
18. Replicate final discs

I. PROGRAM AUTHORING
1. Orient software development team
2. Finalize software design
3. Outline program defaults and standards
4. Log frame #s in program flowchart
5. Author course sequences
6. Develop formative software testing
7. Interface software with check disc
8. Pilot test software
9. Log all bugs and inconsistencies
10. Revise software
11. Outline software operation guidelines
12. Replicate software

J. PRINT PRODUCTION
1. Orient graphic designers
2. Orient technical writers
3. Write draft of copy to be used
4. Revise copy to be used
5. Create possible design thumbnails
6. Choose best design(s)
7. Identify photo and design needs
8. Create design roughs
9. Create all necessary graphics
10. Obtain all necessary photos
11. Obtain printing bids
12. Identify typesetter and printer
13. Set type for all copy
13. Scan or paste-up photo/graphics
14. Obtain comprehensive copy
15. Pilot the print materials

Figure 8. Interactive Design Management System model (continued)

Interactive Design Management System Model

J. PRINT PRODUCTION (continued)
16. Revise and review print materials
17. Reproduce print materials

K. EVALUATION
1. Develop program evaluation documents
2. Review program evaluation documents
3. Revise program evaluation documents
4. Create pre-tests
5. Review pre-tests
6. Revise pre-tests
7. Create post-tests
8. Review post-tests
9. Revise post-tests

Figure 8. Interactive Design Management System model (continued)

Phase 4- Software System Development

Once the IDMS model was constructed, the development of a software-based system for managing the design process was initiated. This process took place in three stages: (a) software design, (b) software production, and (c) software documentation. The following sections describe the procedures that took place during each of these development stages.

Software design. The software design was based upon the 191 tasks defined within the IDMS model. One hundred and eighteen tasks which required the recording or processing of design-related information were identified. Next, software tools were identified that would best facilitate the processing of the design information. Practitioner's notes and comments related to software tools were also referenced during this tool identification process. The list of tools was then analyzed to define interactive paths between the tools. These interactive paths included any points at which design information is either transferred or cross-referenced. A final set of design management tools was then derived from the software task/tool list. The final design management tool list included: (a) forms for recording instructional design information, (b) project and element management schedules, (c) a tool for creating program flowcharts, (d) forms for recording production and storyboard information, and (e) calculators for estimating videodisc space. The result of this process was a conceptual software tool system that would allow interactivity between each of the tools.

Software development. A pilot version of the software was developed using version 1.0 of Apple's Hypercard authoring software (Atkinson, 1987). This pilot version included tools for flowcharting, storyboarding, scripting, and videodisc space calculation. During the development of these tools, a large number of obstacles were encountered due to the constraints of the authoring software. Because of the severity of these constraints, the Hypercard software development was discontinued. The following list summarizes the major constraints imposed Hypercard.

- Nonstandard user interface -- could not create dialog boxes or menus
- Only one tool (window) could be open at a time -- restricted cross-referencing.
- Limited card size -- would not allow the use standard 8.5" by 11" pages.
- Graphic tool limitations - could not monitor use of graphic tools.

A newly released authoring system, Silicon Beach's SuperCard (Appleton, 1989) was then adopted. SuperCard was not restricted by the constraints imposed by Hypercard. Development was begun on a SuperCard version of the software.

Midway through the SuperCard software development, an Alpha version of the IDMS system was demonstrated to two IV practitioners. During this demonstration, the practitioners identified a number of constraints the design of the software imposed on the IV design process. The following is a list and description of the major constraints identified during the demonstration.

- Too information-intensive -- the program required an extreme degree of design detail.
- Restricted design process flexibility -- the program forced the user to follow one predetermined design process. Certain tasks had to be completed before others could be initiated.
- Too computer-intensive -- the program required that all design work be done on the computer. Non-Macintosh and noncomputer user's needs were not addressed.

After considering these constraints, the IDMS software was modified to allow for a greater degree of flexibility in the design process. Instead of requiring the user to fill out a series of forms related to analysis and the instructional design, a computer-based book of questions was created to encourage, not force, the user not to overlook important tasks and issues. The design tools were redesigned so that they could be used independently or connected to interactively, depending on the user's needs. To avoid the creation of a completely closed system, features were added to allow the user to import and export text as well as launch external applications from within IDMS. Capabilities were also added to allow the system to produce blank forms and templates, so non-Macintosh user's could record design information without using the computer. The resulting version of the IDMS software allowed the user greater flexibility to edit and integrate the tools that best fit their unique design process. The design and specific capabilities of the IDMS software are described in detail in the IDMS software manual contained in Appendix D.

Software documentation. Once the Beta version of the IDMS software was completed, an illustrated user's manual was developed (see Appendix D). This manual was designed to serve as an introduction and reference to using the IDMS software. The contents of the manual included information related to: (a) IDMS capabilities and limitations, (b) the application and design of IDMS, (c) installation of IDMS, (d) then use and operation of each software tool, (e) suggested work procedures, (f) the IDMS process model. A detailed index was developed with the manual to facilitate its use as a reference. The included illustrations were taken directly from the software to increase a reader's familiarity with the look of IDMS tools, even when away from the computer. The manual was duplicated and spiral-bound with a diskette pocket for software distribution and testing.

Phase 5- Software Evaluation

During this final phase, the utility and design of the IDMS software was evaluated by three IV practitioners. The practitioners were contacted from the list of IV design experts identified prior to the Practitioner Model Collection, Phase 2. None of the selected practitioners had participated in the model collection.

A meeting was arranged with each of the practitioners in which the IDMS software was explained and demonstrated. Each practitioner was then given a formative evaluation instrument and asked to spend a minimum of two hours working with the tools provided within IDMS. The formative evaluation contained a four-page matrix listing 16 suggested tasks to encourage the practitioners to use each of the tools (see Appendix E). During the evaluation period the practitioners were asked to note on the instrument any problems or frustrations they experienced relative to each of the suggested tasks. This information was latter used to identify and redesign problem areas within the software and documentation.

Once each practitioner had completed the formative evaluation of the software, they were asked to complete a summative evaluation of the overall utility and design of the IDMS software (see Appendix F). The summative evaluation contained a series of 10 questions that were derived from Research Questions 3 and 4. These questions focused on the designer's evaluation of the efficiency and accuracy of the system for managing the design of IV programs, and systems utility as an instructional tool. Each question required the evaluator to make a "yes" or "no" response and provided a space for further comments. The following section first outlines the different backgrounds of the evaluators and then summarizes their responses to each of the summative evaluation questions.

Evaluator backgrounds. The three practitioners who participated in the evaluation work within organizations that design IV programs from different perspectives. David Shapiro is a designer/developer within a small IV development group within Applied Learning, an international producer of education and training programs. His main responsibility is to create and script workable designs from predescribed content. He is an expert Macintosh user, and he uses a computer data base to produce and manage design information. Dr. Eugene Grossman is a professor at the University of Minnesota, Duluth, and has managed teams of part-time student workers in the production of IV programs. In addition, he has taught a series of university courses on IV program design and development. At the time of the evaluation, Dr. Grossman had not used a Macintosh computer. However, he had used a variety of PC-based project management and data base programs to manage earlier projects. Gene Ramsey is the president and primary developer for Image Premastering Inc., a small company that assists external clients in producing and mastering IV programs. He has designed and produced a wide variety of specialized IV programs for government and medical applications. He uses Hypercard and spreadsheet applications to design and manage projects on the Macintosh. The evaluations of these three experts in context of the 10 summative questions follow. The full script of their evaluations are contained in Appendix G.

Question 1- Will the system increase work efficiency? Two of the evaluators felt that the system would increase work accuracy. Although they felt the task-based approach was a good method for accomplishing integrated activities, it also had some drawbacks. Shapiro was concerned that the tasks involved in using the system itself took a lot of time whereas Ramsey felt it could be too information-intensive for some projects.

Question 2- Will the system increase work accuracy? All of the evaluators agreed that use of IDMS would increase work accuracy. By forcing designers to consider and document design-related details, the system makes it difficult to overlook important details. However, Ramsey also pointed out that some designers may hesitate to work at the level of detail the system allows for.

Question 3- Is the system capable of working within your current work procedure? None of the evaluators felt that the system would fully meet their current design management needs. Although all of the evaluators found some of the tools to be very useful, they found that others were not applicable to their particular situations. Shapiro felt that the system would be more useful within a larger development team that had greater reporting needs. Grossman felt that the tasks were aligned with his development process, but that some of the tools were not sophisticated enough to accommodate multiple-disc projects and the management of part-time staff. Ramsey was satisfied with the system, but did not have use for all the tools it provided.

Question 4- Is the IDMS model complete? Although two of the evaluators felt that the IDMS model was complete, Ramsey felt that there were some delivery system specification issues it did not address. These issues were related to programs incorporating multiple-disc players, use of four-channel audio, and extended-feature disc players. Although Grossman was satisfied with the model, he pointed out that it did not address staff training, morale, and supply-related tasks.

Question 5- Is the system complete? All of the evaluators waivered as to the completeness of the system. Shapiro was satisfied with range of tools, but felt there were external applications that might perform some of the tasks more efficiently. Grossman felt that the system was sufficient for small projects, but it could not support the more sophisticated statistical analysis and management tasks involved with large, multiple-disc projects. Ramsey was also satisfied with the current tools, but indicated a need for additional capabilities. His primary interest was the capability to connect and operate external videodisc players from within the system.

Question 6- Does the system offer any unique design and pilot testing capabilities? There was a fair amount of agreement between the evaluators as to the uniqueness of the system's pilot-testing capabilities. Shapiro and Grossman both felt that there were other design programs that offer similar features, but without the range of tools IDMS provides. Ramsey felt the pilot-testing capability was unique to IDMS, but did not add any further details.

Question 7- Would students learning the IV development process benefit from using this system? All of the evaluators responded the most positively to this application. Shapiro and Ramsey emphasized that this was the software's greatest strength, not only for students, but for anyone who is just becoming familiar with the IV design process. Ramsey showed interest in packaging this type of product for new clients who are starting to work with IV and need to communicate their design ideas. Grossman, the only IV instructor, felt that IDMS was a powerful instructional tool because it is structured and understandable, yet flexible.

Question 8- Which tools and accessories did you use? How useful was each tool? This section of the evaluation was different from the others because it addressed the utility of each tool and accessory individually. Although all of the evaluators used each of the tools and accessories, there was very little agreement on which were the most useful. As intended in the development of the comprehensive model, each evaluator felt a different tool would be most useful, but no two agreed. The accessories received more overall, positive concensus. None of the tools or accessories were indicated to be useless.

Question 9- What is the system's greatest strength? All of the evaluators felt that the systems greatest strength was it's ability to encourage the user to consider all the elements and tasks involved in IV program design. Both Grossman and Ramsey reemphasized its applicability for teaching IV design. Grossman also felt that IDMS was well suited for small IV projects that don not demand the power of more expensive project and design management tools.

Question 10- What is the system's greatest weakness? Although none of the evaluators described the system's weakness in the same way, they all indicated a feeling of constraint. Shapiro felt that IV programs require a highly customized design process and that the combined use of independent, external applications allowed for a higher degree of flexibility. Grossman indicated a need for more sophisticated project management, statistical analysis, and budgeting tools than IDMS provided. He reemphasized that these tools were necessary for large, multiple-disc programs, and when managing a large part-time development team. Ramsey felt that IDMS was too information-intensive for the needs of normal projects.

Summary

In summary, the literature-based model and the practitioner-based models had significant overlap. In general, when they were in conflict, the practitioners prevailed in the development of the comprehensive model.

The software system's development and evaluation followed standard practices. Extensive formative evaluation and testing during software development ensured the operation and technical accuracy of the software. The professional evaluation of the utility and efficiency of IDMS was left to three expert IV program designers. Their general assessments were that IDMS works, that it's greatest strength is for use by novices, and that its greatest weakness is that it may be too extensive for many IV design projects.

CHAPTER V

SUMMARY AND CONCLUSIONS

The purpose of this study was to develop and evaluate a computer-based system for managing the design and pilot testing of interactive videodisc programs.

The first four research objectives focused on the development phases of the study. In order to construct this system, a comprehensive design process model was developed from literature- and practitioner-based design models. This model depicted a specialized set of tasks and tools used to manage the IV design process. The complete model and its development is fully documented in chapter four. From this comprehensive model, a software system was created that facilitated the use and integration of its various components. The software is documented in Appendix D and its is development is documented in chapter four. The resulting Interactive Design Management System (IDMS) was then evaluated by three IV program designers in order to answer the last two research objectives which focused on the evaluation phases of the study. Although all of the practitioners felt IDMS was a workable system, the information collected through this evaluation showed many interesting differences in the practitioners perceptions of the system's design and usage in relation of their individual work processes and procedures.

Evaluation Summary

The conclusions drawn from this final evaluation can be used to respond to the two final research objectives. The first is a question about the system's developmental utility and efficiency. The second is a question about the user-friendliness and instructional utility of the system. The following sections discuss the findings and issues related to these objectives.

Utility and efficiency. The practitioner evaluations and comments showed that IDMS's utility and efficiency varied in relation to the specifics of the situation in which it is being applied. The suitability of a situation depends greatly upon both the established design process and the size of the project and development team.

As evidenced in the previous model-collection phases, individuals and organizations specializing in IV design often develop a customized design process. The emphasis of these custom processes is often attributable to the breadth of the designers internal and external accountability. Because the comprehensive design process model used to identify the IDMS task and tool set was created from a wide variety of design process models from different applications and disciplines, its breadth and level of detail is much greater than most processes that have been customized by designers. Because of this breadth, none of the evaluators felt they would use all of the tools. Instead, each of the designers identified a unique subset of tools that would be the most useful in their established design process. This result was both expected and desired.

IDMS's comprehensive design provided capabilities and options that greatly exceeded most practitioners needs. Because of excess, it was difficult for the evaluating designers to assess the overall utility of IDMS since each found only a percentage of the available tools and suggested tasks to be useful. All of the designers found it easier to evaluate the utility of the specific tools or components contained within the system rather the system itself. Each designer felt that some of IDMS's tools would be very useful in their work, and the design practitioners did not choose the same tools. Again, this result was expected.

Evaluator comments also indicated that the utility of the tools would vary according to the size of the project and design team. Although one of the designers voiced concern over the ability of the system to process the massive amounts of design data required for large projects, there was also concern that some of the tools were too information-intensive for either small or simple projects. These concerns also varied from tool to tool, and there was no agreement among evaluators. This lack of agreement indicates that the effect a project's size has on a tool's utility is also relative to differences between practitioner's established design processes and needs further investigation.

Depending upon work procedures, IDMS's utility can be effected by a project team's size since the system was designed to be used by a limited number of users on one shared computer. Because the software had not been tested on a network, the evaluators felt its utility might be of limited use to a large, integrated design team. One evaluator felt that a design-team manager could use the system to generate and disseminate design data to a large team, but that the scheduling and personnel management tools were not powerful enough to handle the necessary record-keeping. Again, these questions deserve further investigation.

Similar to IDMS's utility, the evaluators had difficulty evaluating the overall efficiency of the system. All of the evaluators felt that some of the tools would increase their work efficiency during specific tasks. However, none of the evaluators felt that adopting the entire system would increase efficiency. This agreement not to adopt the entire system once again pointed out that the evaluators had difficulty working within a environment (system), which provided more options than they require. It was generally felt that IDMS lacked efficiency because of its breadth, even though its users are not forced to use any of the tools it provides. Clearly, the IDMS is more expansive than the immediate work demands presently facing the practitioner/evaluator, and this is viewed as highly desirable.

All of the evaluators had previously used a variety of different nonintegrated software applications rather than working within a system. Even though IDMS is an "open system", which allows the users to easily open and use outside applications, adoption of the system appeared to be intimidating. Two of the evaluators actually spoke of separating (unintegrating) the system's tools into independent applications. IDMS requires experienced designers to make some adjustments to the new work methods.

Although IDMS was designed to increase work efficiency by allowing the user to link related documents together, very little was said about this capability. All of the evaluators' comments indicated that they did not perceive any major benefits to creating and working within an integrated environment. If there are advantages, it may take the user time to realize and reap such benefits.

Friendliness and instructional utility. Above any other benefits, all of the evaluators felt IDMS was an excellent tool for guiding novice IV program developers through the design process. All agreed that this type of comprehensive system provided a friendly, but flexible, development environment. They felt that the tasks and checklists provided within the system gave first-time developers the opportunity to understand the complexity of the design process while offering enough guidance, checks and balances to ensure a successful design. Many of these common-sense checks and balances were felt to be critical for novice developers, but often too information-intensive and time-consuming for the experienced practitioner.

Discussion of Additional Findings

Throughout the entire evaluation phase it was quite apparent that the designers did not see any unique benefit of a comprehensive design tool system. Although practitioners' design models that were collected in phase 2 depicted systematic design processes, none of the evaluating practitioners could perceive of linking design tools to integrate the various components of the design process during development. Because the various design components become so obviously integrated in the end product, why do designers prefer to work with them independently throughout the design process? It appears that the concept of working within a system, even an open one, is often perceived as constrained. Although the creativity necessary for designing IV programs thrives in a flexible, unconstrained environment, accuracy and quality assurance require a greater degree of control. This appears to be a very difficult balance to achieve. If integrated systems are the solution, designers most perceive them to be flexible and nonconstraining. New hardware and software technologies are beginning to make such systems feasible, but work procedures and perceptions must change before the true potential of these systems can be realized.

Although IDMS is actually a subsystem residing within the larger, computer operating system, a truly "open" system would allow integration and communication between all applications residing in a computer's operating system. Ted Nelson (1990) described this type of "Hypermedia" working environment as one in which all information can be shared and referenced back to its source. These capabilities would revolutionize future development and publishing environments. Programs such as Hypercard and SuperCard can be seen as only the beginning of this new breed of software systems. Truly integrated systems should be so transparent that the user will not feel any constraints beyond the computer itself.

Recommendations for Further Research and Development

The designer's overall reaction to the concept of an comprehensive, integrated system raises further questions about IDMS and its applications. These questions involve looking closer at both experienced and novice IV designers, as well as the software itself. These final sections consider a variety of factors that are relevant for future investigation.

Experienced designers. All of the designers came to IDMS with well-established work procedures based upon past projects and a limited number of software applications. Because of this, they felt they would only use a subset of IDMS's tools to fit their work process. If experienced designers did adopt a subset of IDMS's tools, would this subset expand as the designers became more comfortable working within the system? Once the designers actually used a number of the tools, would they begin to use more of the linking and interactive functions the system provides to expedite their work process, thus change their work process?

Many designers work on projects for external clients who are unfamiliar with IV program development. Could IDMS be used as a system for communicating ideas between the designer and the client, or the client and the designer? How would this type of communication affect the required design time and the accuracy of the end product?

In order to fully pilot test the flow and interaction of a program's components with IDMS, designers need to input a large amount of design information. Because the designers were not required to provide this level of program detail during the evaluation phase, this capability was not thoroughly tested. Further investigation is necessary to assess the utility and accuracy of this capability.

Novice designers. Although all of the evaluating designers felt that IDMS was best suited for novice IV designers, this must be tested. If experienced designers feel that the system is too information-intensive, will novices feel comforted by its comprehensiveness or overwhelmed by its complexity? Because novice designers will not come to IDMS with a predetermined design process, will they adopt more or all of the system's tools? Once a novice designer using IDMS gains more experience, will he or she continue to use the system, or will he or she feel it is limiting? All of these issues should be addressed before IDMS is adopted as an instructional tool.

IDMS software. Further in-depth testing must be conducted on the IDMS software before it can be fully adopted within a commercial IV development environment. Because most large development groups would require that the software be accessible to a number of users, this is a primary issue. Other considerations include: Could the program be accurately managed if installed on a number of different computers? How well would IDMS operate in a network environment?

Most large commercial IV projects require the management of massive amounts of data. How well would IDMS hold up under these conditions?

What are its limits? What size project is IDMS best suited for? At what point does the program limited speed become an issue? Since IDMS has not yet been used to develop a IV program, what problems and what benefits remain undiscovered? Does IDMS really contain a comprehensive set of design tools and tasks, or are there significant deficiencies in the system?

And finally, what could be done to improve IDMS? As software capabilities continue to expand, new tools and strategies will become available that will further enhance the IV program design process.

References

Aguilar, R. J. (1973). Systems analysis and design in engineering, architecture, construction, and planning. Englewood Cliffs, NJ: Prentice Hall Inc.

Appleton, B. (1989). SuperCard (Computer program). San Diego: Silicon Beach Software.

Atkinson, B. (1987). Hypercard (Computer program). Cupertino, CA: Apple Computer.

Baker, T. F. (1981). Software design or what stands between requirements and programs. In W. W. Cotter, D. J. Couger, N. L. Enger, & F. Harold (Eds.), System analysis and design: a foundation for the 1980's (pp. 235-251). New York: North Holland.

Banathy, B. H. (1987). Instructional systems design. In R. M. Gagne (Ed.), Instructional technology: foundations (pp. 85-112). Hillsdale, NJ: Erlbaum.

Bertalanffy, L. V. (1968). General system theory: foundations, development, applications. New York: Braziller.

Bork, A. (1989). Production of technology-based learning material tools vs. authoring systems. Instructional Delivery Systems, 3 (2), 22-24.

Braden, R. A. (1986). Visuals for interactive video: Images for a new technology (with some guidelines). Educational Technology, 26 (5), 18-23.

Chapin, N. (1981). Graphic tools in the design of information systems. In W. W. Cotter, D. J. Couger, N. L. Enger, & F. Harold (Eds.), System analysis and design: a foundation for the 1980's (pp.121-162). New York: North Holland.

Cohen, S. L., L'allier, J. J., & Stewart, D. (1987). Interactive videodisc-- then, now, and minutes from now. Training and Development Journal, 41 (10), 31-36.

Conner, M. (1981). Structured analysis and design technique. In W. W. Cotter, D. J. Couger, N. L. Enger, & F. Harold (Eds.), System analysis and design: a foundation for the 1980's (pp. 213-234). New York: North Holland.

Daynes, R. (1984). The videodisc book: a guide and directory. New York: John Wiley.

Daynes, R. (1987). General approach to the development of an interactive video project. Chapter of workbook developed for the University of Minnesota Telecommunication Development Center's interactive video workshop.

DeBloois, M. L. (1982). Principles for designing interactive videodisc instructional materials. In M. L. DeBloois (Ed.) Videodisc / microcomputer courseware design (pp.25-66). Englewood Cliffs, NJ: Educational Technology Publications.

Floyd, S., & Floyd, B. (1982). Planning, implementing and budgeting interactive video programs. In S. Floyd and B. Floyd (Eds.) Handbook of interactive video (pp.107-122). White Plains, New York: Knowledge Industry Publications.

Hawthorne, D. L. (1986). How far should the " concept stage" go? Part one: E-ITV's interactive videodisc focus group series. Educational- Industrial Television, 18 (12), 14-19.

Huntley, J. S., & Alessi, S. M. (1987, July/August). Videodisc authoring tools: evaluationg products and a process. Optical Information Systems, 259-281.

Iuppa, N. V. (1984). A practical guide to interactive video design. White Plains, New York: Knowledge Industry Publications.

Johnson, J. F., Widerquist, K. L., Birdsell, J., & Miller, A. E. (1985) Storyboarding for interactive videodisc courseware. Educational Technology, (12) 25, 29-35.

Kearsley, G. (1988). Beyond authoring: Software tools can help manage program development. Instructional Delivery Systems, (1) 2, 15-20

Kearsley, G. P., & Frost, J. (1985). Design factors for successful videodisc-based instruction. Educational Technology, (3) 25, 7-13.

Konsynski, B. R. (1981). Data base driven system design. In W. W. Cotter, D. J. Couger, N. L. Enger, & F. Harold (Eds.) System analysis and design: a foundation for the 1980's (pp. 252-278). New York: North Holland.

Laszlo, E. (1972). The relevance of general systems theory: Papers presented to Ludwig von Bertalanffy on his seventieth birthday. New York: Braziller.

Lippke, J. A. (1987). Early planning -- finding the answers to all the questions. Part two: E-ITV's interactive videodisc focus group series. Educational-Industrial Television, (2) 19, 26-31.

Lippke, J. A. (1987). Finally, the final plan. Part three: E-ITV's interactive videodisc focus group series. Educational- Industrial Television, (3) 19, 35-40.

Lippke, J. A. (1987). Scripting: more complicated than you might think. Part four: E-ITV's interactive videodisc focus group series. Educational-Industrial Television, (4) 19, 22-27.

Lippke, J. A. (1987). Interactive videodisc production: short segments raise continuity problems. Part five: E-ITV's interactive videodisc focus group series. Educational- Industrial Television, (5) 19, 30-34.

Nelson, T. (1990, March). Establishing the hypermedia work environment. Paper presented at Multimedia New York conference, NY.

O'Bryan, K. G. (1982). Instructional strategy and evaluation. In S. Floyd and B. Floyd (Eds.), Handbook of interactive video (pp.67-84). White Plains, New York: Knowledge Industry Publications.

O'Donnell, J., Harris, D., & Sato, T. (1981) Videodisc program production manual. Sony product communications group publication: New York.

Schwier, R. (1987). Interactive video. Englewood Cliffs, NJ: Educational Technology Publications.

Smith, E. E. (1987). Interactive video: an examination of use and effectiveness. Journal of Instructonal Development, (2) 10, 2-10.

Appendix A

Practitioner Interview Questions

Practitioner Interview Questions

Front End Analysis

1. Do you use objectives? When are they formed? What kind?

2. What types of things do you consider about the audience before the projects onset? How is this documented and communicated? Where is it used?

3. How and when do you define time, staff, and budget requirements?

4. Do you do cost/benefit analysis before beginning?

5. When do you decide what kind of system you will be using to deliver the program?

6. When and how do you define the internal and external evaluation criteria?

Practitioner Interview Questions

Program Design

1. When and how do you decide the level of interactivity?
2. When and how do you decide on the type of input devices to be used?
3. When and how do you define the different types of user interaction that will be required?
4. Do you use flowcharts? For what? At what stages?
5. When and how do you decide on you instructional strategies?
6. When and how do you define you strategies for evaluation and testing?
7. When do you decide on the best presentation medium for a particular segment (still video, motion video, animation, graphics, etc.)?
8. When and how do you define your audio track usage?
9. When do you begin working on the documentation? How is it managed? Is it ever online?
10. How do you plan and control for audio and video transitions?
11. How and when is the disc's personality defined?
12. How do you plan for unfamiliar user operation?

Practitioner Interview Questions

Production

1. Do you consider integrating existing media? When? How?

2. When and how is the video's treatment and style defined?

3. How is computer programming data documented and communicated?

4. How is video production information documented and communicated to the crew? Storyboards? Shot lists?

5. How is audio production information documented and communicated?

6. How is graphic production info documented and communicated?

7. What info is used in post production? Edit decision lists?

Practitioner Interview Questions

Management

1. Do you develop a systematic plan of action? How and when is it updated?
2. How and when do you define personnel responsibilities? How do you monitor their progress?
3. What types of group communication strategies are used? What kinds of reports?
4. How is your budget estimated and monitored through out the project?
5. How is disc space managed and estimated?
6. What type of information do you send out for production bids?
7. When do you review the content before continuing? Check points? Sign offs? How is this evaluation done?
8. How do you monitor group menders changes to the content (script, questions, design, etc.)?
9. Do you pilot test before disc duplication? How?
10. How do you manage your production schedule?
11. What are typical communication problems you have run into in past projects? Can you see any solutions?
12. Do you use any computer-based information management systems? What functions do they perform?

Appendix B

Summary of Models Collected from the Literature

Summary of Models Collected from the Literature

FRONT END ANALYSIS	CONTENT OUTLINE	PROGRAM DESIGN	PRODUCTION REQUIREMENTS	PROJECT MANAGEMENT
Statement of problem	**Identify SME's**	**Interactivity**	**Existing media resources**	**Systematic plan of action**
Objectives	**Task analysis**	Level of interactivity	**Treatment and style**	**Assign responsibilities**
Broad objectives	**Content structure analysis**	Types of interactivity	**Computer production info**	**Communication strategies**
Behavioral objectives	**Learning hierarchies**	Input devices	**Video production info**	**Budgeting strategy**
Needs assessment	**Instructional modules**	Flowcharting	**Audio production info**	**Disc space management**
Audience identification	**Test items and locations**	**Strategies**	**Select talent**	**Production bids**
Audience locations	**Outline entire program**	Instructional strategies	**Select production location**	**Content review strategies**
Learner characteristics	**Content review by SME**	User control strategies	**Identify vendors**	**Pilot testing strategies**
Learning environemt	**Produce script**	Evaluation strategies		**Production schedule**
Resources		Media selection		
Organ. Commitment		Online documentation		
Detemine budget		Audio track usage		
Time requirements		**Consistency**		
Staffing requiorements		Audio transition consistency		
Cost/benifit analysis		Video transision consistency		
Assess system capabilities		Disc personality		
Evaluation		System operation intro		
External evaluation criteria				
Internal evaluation criteria				

BEST COPY AVAILABLE

Appendix C

Original Practitioner Model Documentation

Videodisc Production Plan

PROJECT		VIDEO	Writer	DESIGN	SCRIPT	GRAPHICS	PRINT	STUDIO	LOCATION	EDIT	PROGRAM	PROOF	REPLICATE
[project name]	[lesson/unit name]	[name]	[name]	[date]	[date]	[date]	[date]	[date]	[date]	[date]	[date]	[date]	[date]

Videodisc Production Plan

PROGRAM	WEEK ENDING:	06-Feb	13-Feb	20-Feb	27-Feb	05-Mar	12-Mar	19-Mar	26-Mar	02-Apr	09 Apr	16Apr	23-Apr
Review Script & Flowchart													
Code special routines													
Code logic													
Log disc													
Add frame numbers													
Logic test													
Beta test													
Team Review													
Revise													
Transmit (2)													
Test proof disc (2)													
Re-transmit (2)													
Replication													

Videodisc Production Plan

EDIT	WEEK ENDING:	06-Feb	13-Feb	20-Feb	27-Feb	05-Mar	12-Mar	19-Mar	26-Mar	02-Apr	09 Apr	16Apr	23-Apr
Log source tapes													
Disc Architecture													
Create EDL													
Timecode Calculation													
Primary Edit													
Review 3/4" Dupe													
Clean-ups													

Videodisc Production Plan

LOCATION	WEEK ENDING:	06-Feb	13-Feb	20-Feb	27-Feb	05-Mar	12-Mar	19-Mar	26-Mar	02-Apr	09 Apr	16Apr	23-Apr
Talent Selection													
Site Selection													
Shoot													
Window burn work tapes													
Obtain library footage													

Videodisc Production Plan

STUDIO	WEEK ENDING:	06-Feb	13-Feb	20-Feb	27-Feb	05-Mar	12-Mar	19-Mar	26-Mar	02-Apr	09 Apr	16Apr	23-Apr
Production Design - design environ. - viewer response - production values													
Set Design													
Set Construction													
Props													
Set-up & Staging													
Talent Selection													
Script Read (time & visuals)													
Shot list													
Prompter Scripts													
Shoot													
Window burn work tape													

Videodisc Production Plan

PRINT	WEEK ENDING:	06-Feb	13-Feb	20-Feb	27-Feb	05-Mar	12-Mar	19-Mar	26-Mar	02-Apr	09 Apr	16Apr	23-Apr
Select Document Type													
Design Format													
Write													
Review													
Revise													
Field Test													
Graphic Design													
Typeset													
Keyline													
Proofread galley proofs													
Revise													
Final Review													
To Printer													

Videodisc Production Plan

GRAPHICS	WEEK ENDING:	06-Feb	13-Feb	20-Feb	27-Feb	05-Mar	12-Mar	19-Mar	26-Mar	02-Apr	09 Apr	16Apr	23-Apr
Style/Format Developed													
Storyboard (s) Drawn													
CG Text Frames Collected													
Graphic Design Review													
Audio Recording													
Audio transfer													
Pre-Pro Animation													
Animation Production													
Transition Graphic Prod													
Still Frame Production													
Transition Review													
1st Still Review													
1st Animation Review													
Still Correct/Revise#1													
Anim. Correct/Revise#1													
2nd Still Review-3/4"													
2nd Animation Review													
Correction/Revision #2													
Create sub-master													
Review Transfer-3/4"													
Correct/Revise #3													
Edit Changes to Sub													
Edit Ready to Sub-master													
Produce Opening													

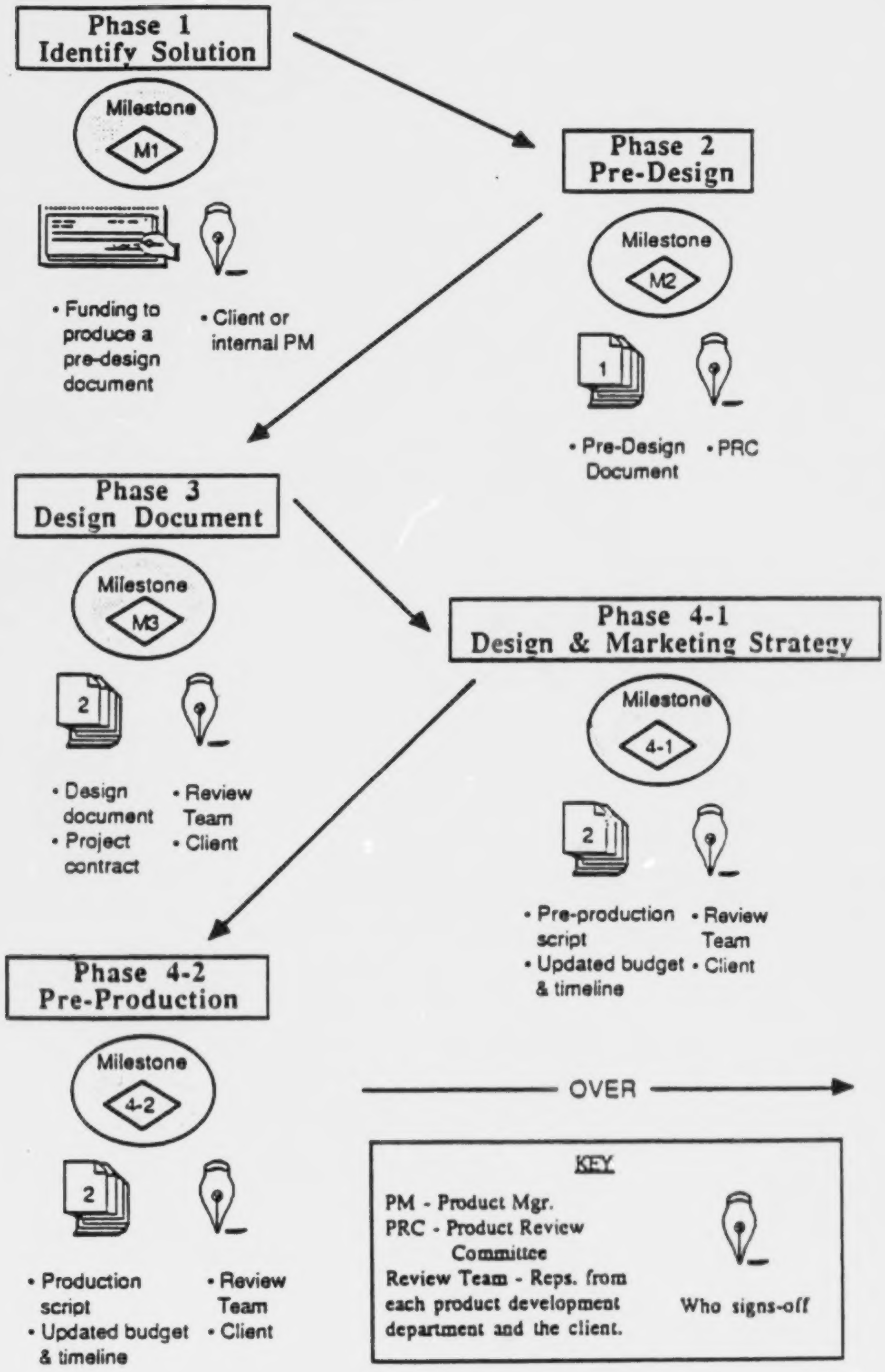
Videodisc Production Plan

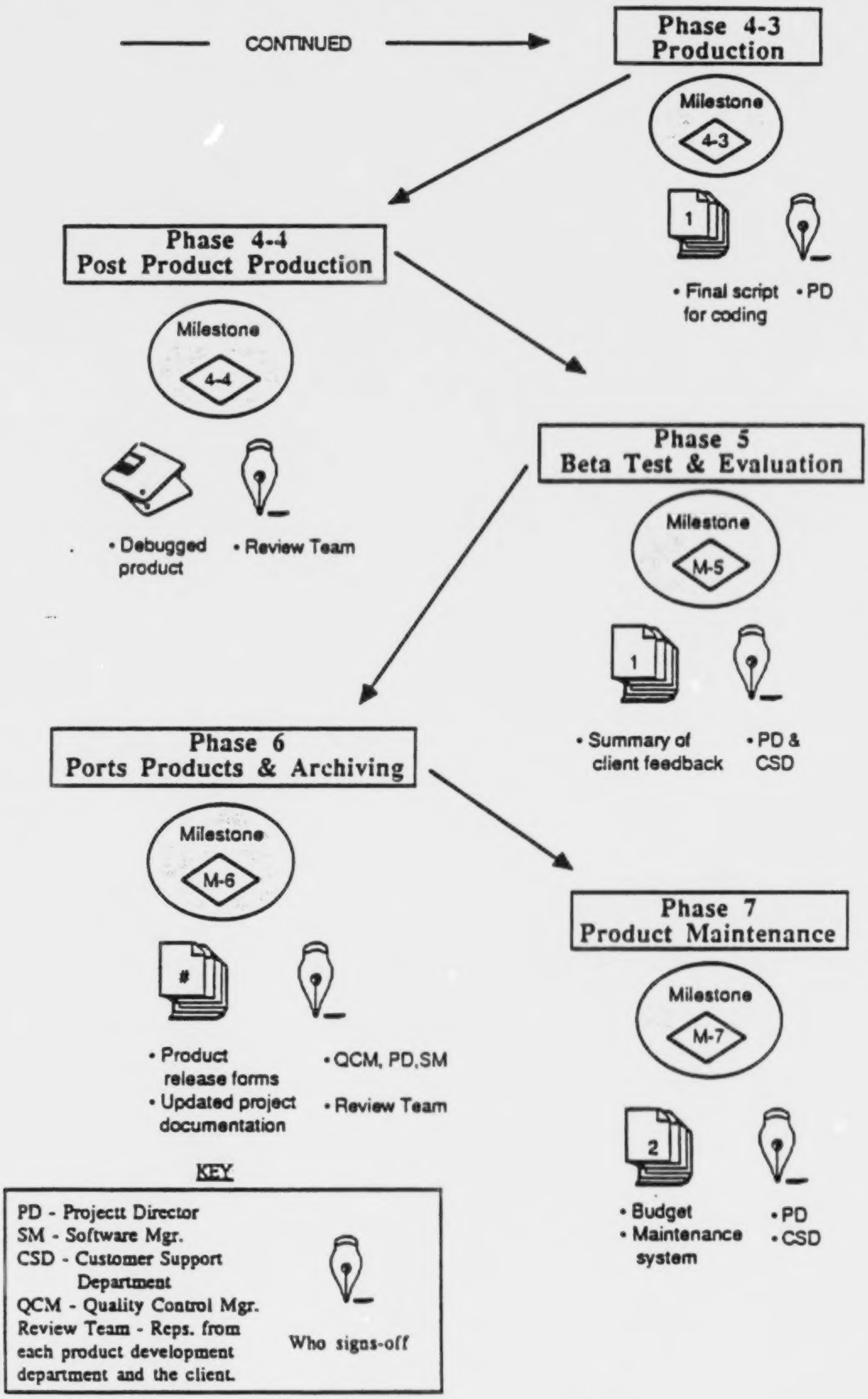
SCRIPT	WEEK ENDING:	06-Feb	13-Feb	20-Feb	27-Feb	05-Mar	12-Mar	19-Mar	26-Mar	02-Apr	09 Apr	16Apr	23-Apr
Content outline													
1st draft													
Team review													
Final													
Detail Flowchart													

Videodisc Production Plan

DESIGN	WEEK ENDING:	06-Feb	13-Feb	20-Feb	27-Feb	05-Mar	12-Mar	19-Mar	26-Mar	02-Apr	09 Apr	16Apr	23-Apr
Task Analysis													
Needs Analysis													
Scope and Sequence													
Concept Flowchart													
Instructional Strategies													
Lesson Design													
Assess Student Performance													
Advisory Review													
Design Summary													

IVD PRODUCT DEVELOPMENT MILESTONES


Phase 1
Identify Solution
Milestone
M1
• Funding to produce a pre-design document
• Client or internal PM
Phase 2
Pre-Design
Milestone
M2
1
• Pre-Design Document
• PRC
Phase 3
Design Document
Milestone
M3
2
• Design document
• Project contract
• Review Team
• Client
Phase 4-1
Design & Marketing Strategy
Milestone
4-1
2
• Pre-production script
• Updated budget & timeline
• Review Team
• Client
Phase 4-2
Pre-Production
Milestone
4-2
2
• Production script
• Updated budget & timeline
• Review Team
• Client
OVER
KEY
PM - Product Mgr.
PRC - Product Review Committee
Review Team - Reps. from each product development department and the client.
Who signs-off


CONTINUED
Phase 4-3
Production
Milestone
4-3
1
• Final script for coding
• PD
Phase 4-4
Post Product Production
Milestone
4-4
• Debugged product
• Review Team
Phase 5
Beta Test & Evaluation
Milestone
M-5
1
• Summary of client feedback
• PD & CSD
Phase 6
Ports Products & Archiving
Milestone
M-6
#
• Product release forms
• Updated project documentation
• QCM, PD,SM
• Review Team
Phase 7
Product Maintenance
Milestone
M-7
2
• Budget
• Maintenance system
• PD
• CSD
Who signs-off

KEY

PD - Projectt Director
SM - Software Mgr.
CSD - Customer Support Department
QCM - Quality Control Mgr.
Review Team - Reps. from each product development department and the client.

IVD PRODUCT DEVELOPMENT PROCESS & MILESTONES

Key

PM - Product Manager
AM - Account Manager
PRC - Product Review Committee
PD - Project Director
SME - Subject Matter Expert
D/WM - Design/Writing Manager
EM - Editorial Manager
PS - Premastering Specialist
SM - Software Manager
GM - Graphics Manager
QCM - Quality Control Manager
CSD - Customer Support Department
DESIGN TEAM - PM, PD, D/WM, SME, SM, GM, PS, QCMz
REVIEW TEAM - PD, D/WM, SME, Co-developer, SM, GM, Marketing Representative, QCM.

Preface

This document is a guide for all the steps necessary from the need for a new product to the delivery and implementation of the product. The steps outlined in this guide are not necessarily performed sequentially; some can be handled simultaneously. Since team members and the number of projects will vary, it is up to the project team to determine when the steps will be carried out. This also represents an "ideal" system for product development. Some steps may be eliminated due to the realities of time and money. It is advisable to do each step as thoroughly as possible. This will help reduce the need to loop back as mistakes are caught or design changes are made.

Curriculum Team Identifies Market Need

A. Discover business problems/market opportunities - PM, CSD, Clients
B. Evaluate opportunity - PM, AM
  Discuss with Product Review Committee
  Resolve legal issues:
    -funding
    -ownership (if client/co-developer)
C. Study the business problem - PM, Marketing
  Estimate the size of the market, business potential
  Read relevant literature
  Consult with colleagues, field, clients

IVD PRODUCT DEVELOPMENT PROCESS & MILESTONES

        Estimate competitive position
    D. Official Approval - PM, AM
        Approval to proceed from WLC Product Review Committee
        Obtain signed letter of agreement (client or internal contract)
        Approve plan (requirements and deliverables)
        Authorize budget for Mkt research and design

MILESTONE DELIVERABLES - Funded approval to proceed with plan for Market Research from PM and AM.

Curriculum Team Researches Business Problem

- A. Execute plan for research - Design Team
    - Obtain project number, incorporate pre-project costs
    - Recruit research/design team, brief them
    - Kickoff mtg, input on project and schedule
    - Publish schedule (refine as needed)
- B. Conduct mkt. research (focus groups, structured interviews, literature search, questionnaires ...)
    - Identify factors affecting performance
    - Determine causes of business problem-Knowledge/skill deficiency? Need for feedback/reinforcement?
    - Organizational barriers?
    - Technology/Hardware considerations
    - Analyze findings and clearly define business problem
    - Confirm that a consulting/training solution is appropriate and worth pursuing

MILESTONE DELIVERABLES - document of affirmation that a market opportunity either exists or does not exist.

Phase 1 Curriculum/Product Dev't Team Identifies Solution

- 1A. Plan integrated offering - PRC, D/WM
    - Define training components, desired results.
    - Identify target audience profile.
    - Define purpose of each solution and desired outcome.
    - Define program user prerequisites
    - Determine sequence of Learning Events.
    - Consider implications for all WLC divisions.
    - Market Survey, will they buy and what will they pay?
    - What format preferences do they have?
- 1B. Develop a plan to obtain design funding
    - Do work breakdown and design budget for the design portion of

each component of solution.
Analyze each component vis a vis competition.
Prepare presentation and present it.

1C. Mgmt team evaluates and prioritizes next steps.
Approve relevant budgets, assign Product Managers.

M1 - MILESTONE DELIVERABLES - Client or internal funded approval to proceed with a product requirement document and creation of pre-design document.

Phase 2 Pre-Design

2A. Execute a plan for design:PM
- Identify resources/designers.
- Recruit and contract resources.
- Brief resources.
- Kickoff mtg with project team to exchange information
- Meet with research team and get latest relevant info.
- Obtain agreement on maintenance of systems files and sharing structures and document the agreement
- Develop and document schedule

2B. Develop a conceptual design framework for each component.- Design Team
- Identify content
- Explore hardware/software delivery issues and implications
- Document specific learning and performance objectives
- Develop a testable design flow that addresses learning objectives
- Test above with an internal audience, evaluate and refine. (repeat until working)
- Review Team Input for design document

2C. Produce a Design document which addresses:
- Purpose and scope of training
- Description of target audience
- Task analysis
- Measurement and evaluation (criterion test)
- Learning objectives, instructional design
- Activities and desired outcomes
- Delivery sequence
- Video requirements and scenarios
- Graphic design (see graphics manual)
- User interface
- Deliverables (with specifications)
- Tailoring plan
- Marketing & customer support requirements

IVD PRODUCT DEVELOPMENT PROCESS & MILESTONES

- Hardware/software requirements
- Print Collateral options
- Testing Strategies

2D. Conduct internal quality review and revise as needed

2E. Develop plan for approval - PD
- Budget requirements (per unit)
- Estimate resource, budget, timeline requirements
- Consult with Marketing on the business plan (is cost justified by the learning component?)
  - -marketing stategy
  - -pricing and estimated ROI
  - -prospect target list
- Prepare and present plan to top management

2F. Approve business plan, authorize budget for development/ production/ marketing

2G. Develop and set responsibility for quality standards according to the above decisions-PM

2H. Trademark unique components (titles etc.) with legal dept.-PD

M2 - MILESTONE DELIVERABLES - Review Team approved pre-design document. (Possibly client funded approval to proceed with design document as well.)

Phase 3 Design Document

Develop:

3A. Design principles - Design Team
- Instructional Design
- Engine outline
- "Treatment" flowchart
  - -content orientation for writing
  - -visual orientation for graphics
  - -evaluate project in light of current sw structures
    - a. evaluate engine design and document new boundaries if needed
    - b. evaluate nodal design and document new boundaries if needed
    - c. evaluate hardware implications and design and document new driver boundaries if needed
    - d. explore new ideas within engine/node/ drivers/ user interface/ and delivery environment boundaries and document
  - -design delivery environment: menu, install, supervisor/ facilitator utilities, etc and document

IVD PRODUCT DEVELOPMENT PROCESS & MILESTONES

- Major content assumptions
    - learning points to cover
- Major product assumptions

3B. Specifics

- Standards/Specifications documents
    - disc real estate
    - video art
    - video text
    - rtg art
    - rtg text
    - transitions
    - production values
    - visual design/color pallette
    - print collateral
    - product & module titles
    - hardware and memory requirements
    - hardware diagrams
    - page numbering system
    - page type naming system
    - software syntax for branching
    - change and/or update approval
    - change and/or update informing system
    - tear open/other liscence agreement-legal
- Writers specification manual - D/WM

3C. Production

- Writing
- Write treatments - D/WM
    - official product description - SW/CSD
    - edit product description - EM
    - outline of learning points-SW,SME
    - edit outline - EM
    - description of content presentation (rolling video, text frames etc)
    - branching instructions

3D. Project Direction

- Refine budgets and timelines
- Get team and co-developer sign off

M3 - MILESTONE DELIVERABLES - A Review Team approved design document which details specific work to be completed with budget and timelines attatched. A client/internal contract to proceed with production.

****Past this point, a change order is needed for any design changes, either additions or deletions.**

IVD PRODUCT DEVELOPMENT PROCESS & MILESTONES

Phase 4-1 Design and Marketing Strategy

- 4-1A. Writing - D/WM
  - Write Draft #1 Scripts to standards and Specifications
    - -electronic script page
    - -audio/video dialogue with rough timings
    - -text
    - -dve descriptions
    - -rtg directions
    - -pc generated sound directions
    - -stage directions
    - -locations descriptions
    - -character descriptions
    - -fill in draft fields: mod, date, text, pg#, script, same as, character, location, type (will encompass disc, bkg, dve, fmt,pmt).
- 4-1B. Review Team review Draft #1
- 4-1C. Overall project structure based on draft #1 scripts - SM
  - Document sw related notes
  - Design data flow for overall project: bookmark, record keeping, etc and document.
  - Finalize engine/nodes decomposition
  - Design new data file structures and identify compatability issues and document
  - Design new hardware-related structures code overlays, file systems etc. and document
  - Evaluate system compatibilities and identify needs for new coding and data file creation/editing and document (iterate)
  - Review software logic (branching from/to, and logic fields as input by writers)
- 4-1D. Marketing
  - Execute Marketing Strategy
    - -develop marketing materials, fact sheet - GM
    - -edit fact sheet - EM
    - -pre-announce product idea to salesfield, give only anticipated quarter for release.
    - -Itemize all product components for Inventory number assignment
    - -establish unit costs for all components and assign parent numbers.
    - -establish hardware configurations
    - -initial Bill of Materials and inventory numbers if product will be carried in inventory.

IVD PRODUCT DEVELOPMENT PROCESS & MILESTONES

4-1E. Project Direction
    Develop "production" flowchart
        -define structures
        -provide parameters for writing
        -test structure, highlight inconsistencies
4-1F. Writing - D/WM
    Write Draft #2 scripts
        -electronic script page(update fmt if necessary)
        -audio/video dialogue with realistic timings
        -vo audio, both tracks with realistic timings
        -text
        -dve use as per design document
        -rtg use as per design document
        -pc generated sound direction consistency
        -stage direction consistency
        -locations description consistency
        -character description consistency
        -draft field consistency: mod, date, disc, bkg, text,
        dve, fmt, pmt, pg#, script, same as, character,
        location.
4-1G. Review Team reviews draft #2
4-1H. Software follows all steps as in 3 above after draft #2 scripts -SM
4-1I. Project Direction
    Modify "production" flowchart
        -define structures
        -cement global calls
    Update timelines and budgets
        -either justify timeline/budget changes or
        -modify scripts to fit initial plan
4-1J. Writing -D/WM
    Continue draft/review stage as necessary
4-1K. Editorial review of final script EM
4-1L. Review Team - review final draft, finalize changes made in editorial.
4-1M. Graphics
    Design Print Collateral materials
Note: see "Graphics Print Component List" and decide on all elements.
    Design Video Art, Video Text, RTG Art, RTG Text
        -initial design
        -test design
        -revise design
        -final design
        -proof design
4-1N SW Eng
    Finalize engine design and walk through with data dictionary and

IVD PRODUCT DEVELOPMENT PROCESS & MILESTONES

dataflow diagram

Finalize nodal design and walk through with data dictionary and dataflow diagram

Finalize hardware drivers design and walk through with data dictionary and dataflow diagram

Finalize data-files structures design and walk through with data dictionary and dataflow diagram

Identify required modifications to current development tools (eg. menugeni, paintbox) and design and document alterations

Create a simulation of the program and test engine and design

4-1O. Customer Support/Marketing

Alpha Test (This phase has never been done for our interactive courseware and should be done with an instructional designer and someone who knows how to Alpha Test. This phase would end the business of continual redesign efforts throughout the product development process and give us feedback at a phase where we can redesign before it gets too expensive!)

Rewrite script based on Alpha feedback

Review team sign-off and commit to design

Product components stock
- -number of laser discs
- -number of floppy diskettes
- -number of packaging materials

4-1P. Quality review of design for all materials before production begins

M4-1 - MILESTONE DELIVERABLES - A Review Team approved preproduction script. A client/internal approval of any budget and timeline changes.

Phase 4-2 Pre-Production

4-2A.Design

Review by PD, MPD, PDGL,D/WM and SME

4-2B.Audio/Video

Review with producer/director, gfx art director, PD,MPD

Break out script

Set and lighting design

4-2C. Editorial-EM

Copyediting
- -content continuity
- -syntax
- -grammar check
- -spelling and proof

Integrated feedback from reviews

IVD PRODUCT DEVELOPMENT PROCESS & MILESTONES

    Proof script with "production" flowcharts and Group Standards/Specifications document
- 4-2D. SW engineers - SM
    - Review any changes to software syntax since draft #2
- 4-2E. Graphics - GM
    - Proof gfx fields (input by writers)
    - Complete script break out
    - Review set and lighting design
    - Begin review of print collateral vendors

M4-2 - MILESTONE DELIVERABLES- A Review Team approved production script and client/internal sign-off on any design, budget or timeline changes.

Phase 4-3 Production

- 4-3A. Audio/Video - PS
    - Complete production
        - -create continuity logs
        - -create video/film logs
        - -record in Production Master. Script:
            - a. script changes - D/WM
            - b. actual timings
            - c. best takes
            - d. source reel numbers
            - e. actual talent data
    - Produce VO Audio
- 4-3B. Graphics - GM
    - Video art and logs
    - Proof video text
    - Vidifont text and logs
    - Proof Vidifont text
    - RTG art
    - RTG text
    - Proof RTG text
    - Produce print collateral (except content of guides and workbooks.
- 4-3C. Premaster - PS
    - Duplicate audio/video masters
    - Edit off-line
    - Build DVE's
    - Duplicate VO audio masters
    - Edit VO audio
    - Transfer audio to 1" tape
    - Assemble source reels

IVD PRODUCT DEVELOPMENT PROCESS & MILESTONES

        -video art
        -video text
        -stillframe build
4-3D. Project Direction - PD
    Input sequence map
        -sort
        -check
    Create/define music
        -original
        -library
    Complete "assembly" flowchart
        -document actual production additions/changes
4-3E. SW eng - SM
    Assign coding tasks for engine/nodes/drivers/delivery environment
    Input data into menugeni, paintbox etc
    Code engine and modules and document (hardcopy sufficient)
    Code machine dependant and independant hardware drivers (where necessary) and document (hardcopy sufficient)
4-3F. Writers/Editorial - D/WM
    Collateral materials composed,edited, revised in Marketing, tested in QC, input and handed off to graphics in electronic form.
        -workbooks
        -co-ordinator's guide
        -salesperson guides
        -implementation handbooks
4-3G. Graphics - GM
    Design collateral materials

4-3H. Review Team - sign off, script enters ASSEMBLY
    (release minimum of 1 week before on-line)
4-3I. Proj Direction
    Check source reels
4-3J. Premaster
    Revision
    Edit on-line
        -edit numbers
        -SMPTE time code
        -editor comments
        -master reel numbers
    Sweeten master tape
        -source materials
    Master 1" tape review with MPD
    Revisions to 1" master tape
    Duplicate 1" master tape

IVD PRODUCT DEVELOPMENT PROCESS & MILESTONES

Proof disc or Master disc and press replicate

4-3K. SW Eng
  Finish coding
4-3L. Proj Direction
  Create final flowchart
  Review disc for mastering problems
  Create disc map log
4-3M. Project Director signs off.

M4-3 - MILESTONE DELIVERABLES - A PD approved final script for final coding.

Phase 4-4 Post Product Production

4-4A. SW Eng
  Adjust and complete frame-related data files
  Integrate system and test (debug)
4-4B. QC Test/recommend/recycle through editorial &/or writing
  Software
  Collateral
  Design
[Loop through sw engs as needed]
4-4C. Graphics
  Supervise scheduling for production of print collateral.
  When QC of collateral is final, replicate.
  Proof quality of print collateral.
4-4D. Project direction
  Authorize revisions until complete
4-4E. Sign off -review team

M4-4 - MILESTONE DELIVERABLES - release of debugged product to CSD for Beta Testing. Review Team sign-off of product.

Phase 5 Beta Test and Evaluation

5A. CSD Registers product
5B. Completes final product check
  Legal criteria met
  Collateral materials complete
  Item and hardware numbers in place
  Marketing information available
5C. Replicate and assemble for Field Test
  -Discs, software, collateral
5D. Design Beta Test and identify test sites

IVD PRODUCT DEVELOPMENT PROCESS & MILESTONES

5E. Ship and Client Field test
    Possible rework and QC based on field feedback.
    Develop support procedure for field bugs and enhancements
    Release final product
    Mass Replicate and Assemble Stock
    Sell
    Ship orders

M5 - MILESTONE DELIVERABLES - A summary document of client feedback from CSD and PD sign-off for any product updates requested.

Phase 6 Ports, Product Releases, and Archiving

Ports
6A. Schedule secondary ports once initial feedback from Beta Testing is in and no major problems are indicated.
Product Releases
6B. Schedule release of product on different hardware configurations.
Archive
6C. "Originals" storage
    Paper script/logs
    Electronic script/logs
6D. Premaster
    Original glass master storage
    Original source reels
    Original 1" master
6E. SW eng
    Floppy/back up tape storage
6F. Review Team holds post mortem on techniques, costs and timelines, and document what was learned.

M6 - MILESTONE DELIVERABLES - A complete set of product release forms signed by PD,SM, and QCM for each port, all elements of the project safely stored and accessable as needed, a document summarizing what what learned and all system documents updated to reflect the structure to be used for the next project.

Phase 7 Product Maintenance

7A. System for feedback input to a database or to SM for terminal/critical bugs. PD,SM
7B. Product updates or re-releases. SM
7C. Testing portability to new hardware systems. SM
7D. Testing life of components, tracing problems related to videodiscs or

floppy diskettes. PD
(There may be two separate budgets since CSD functions should be ongoing for the life of the product.)

M7 - MILESTONE DELIVERABLE - WLC IVD products have the reputation of being supported in the field.

Project Requirements Document

PROJECT DEMOGRAPHICS

DATE>> ☐
PROJECT #>> ☐
PROJECT NAME>> ☐
GENERAL DESCRIPTION>> ____________

PROJECT DIRECTOR>> ☐
DESIRED ROLL-OUT DATE>> ☐
# OF DISC SIDES per versions)>> ☐
TOTAL # OF SIDES (all versions)>> ☐
TOTAL # OF MODS. (all versions)>> ☐
# OF MODULES PER VERSION>> ☐
NUMBER OF VERSIONS>> ☐

DESCRIPTIONS>> ____________

TYPE OF PROJECT ____________

INTEGRATED STRATEGY ____________

DELIVERY MODE (select only one with "X"):

IVD>> ☐
or
CBT>> ☐
or
OTHER>> ☐ **DESCRIPTION>>** ____________

TYPE OF PROJECT (select only one with "X", then fill in its sub-choices):

GENERIC>> ☐

or

SPEC. DESIGN>> ☐

or

TAILORING>> ☐ **ORIGINAL COST>>** ☐
BASE PROG. NAME>> ____________

Project Requirements Document

CONTENT (all modules):

% OF VIDEO ALREADY EXISTING>>	
NUMBER OF LEARNING POINTS>>	
HOURS OF LEARNER TIME>>	

PRE-ASSESSMENT (Y/N)>>	
POST-ASSESSMENT (Y/N)>>	
PRE-SELFTEST (Y/N)>>	
POST-SELFTEST (Y/N)>>	

PORTION FUNDED BY CLIENT (all modules):	$	%
PRODUCT (fill in one)>>		
SOFTWARE MAINTENANCE (fill in one)>>		
ENHANCEMENTS (fill in one)>>		

	%
PERCENTAGE OF WLC OWNERSHIP (all modules)>>	

PERSONNEL REQUIREMENTS

PROJECT DIRECTION:	#	DAYS	DAY RATE	AMT.
PROJECT MANAGER>>				
PROJECT COORDINATORS>>				
SOFTWARE PROJ. MGR.>>				

OTHER PROJECT STAFF:		
SUBJECT MATTER EXPERTS>>		
INSTRUCTIONAL DESIGNERS>>		
INTERACTIVE DESIGNERS>>		
WRITERS>>		
SUPPORT STAFF>>		

HARDWARE REQUIREMENTS

	number	1 to 10 degree of complexity	0 to 100 % complete
SUPPORTED TARGET SYSTEMS>>			
UNSUPPORTED TARGET SYSTEMS>>			
MINIMUM # OF DISKETTE DRIVES>>			
MINIMUM # OF HARD DRIVES>>			
MINIMUM MEMORY SIZE (Kbytes)>>			

Project Requirements Document

SOFTWARE IMPLEMENTATION

GAMING TYPES:	number	%complete	cost	amt
WORD MATCHING>>				0
SELF-RATING SCALE>>				0
DVE GAMES>>				0
APPROACH GAMES>>				0
SLIDING BAR>>				0
SLIDE SHOWS>>				0
WORD RANKING>>				0
SIMULATIONS>>				0
MULTIPLE CHOICE>>				0

OTHER INTERACTIVE FEATURES (fill in):	yes/no	1 to 10 average complexity	amt
LEARNER'S CHOICE>>			
INTERRUPT>>			
STUD. REC. KEEPING>>			
ITG ENGINE>>			
INSTALL>>			
MENU>>			
MANAGER'S MOD>>			
PIC FILES>>			

GRAPHIC DESIGN (total all sides)

	number	% complete	Amt.
IMAGES:			
COMPUTER LOGO ANIMATION>>			0
# OF GENERIC BACKGROUND PAGES(MEMOS)>>			0
# OF BACKGROUNDS FOR TEXT &/OR DVE'S>>			0
# OF CUSTOM BACKGROUND PAGES>>			0
VIDIFONT:			
# OF VIDIFONT CUSTOM PAGES >>			0
# OF VIDIFONT TEXT PAGES>>			0
PC GRAPHICS:			
# OF PIC FILES>>			0
# OF CUSTOM PC GENERATED TEXT PAGES>>			0
# OF PC GENERATED TEXT PAGES>>			0

GRAPHIC DESIGN - PACKAGING FOR THE PRODUCT

	yes/no	1 to 10 degree of complexity	0 to 100 % complete	Amt.
PRODUCT FACT SHEET>>				0
VIDEODISC ALBUM>>				0
5 1/4" SOFTWARE FLOPPY LABELS>>				
3 1/2" SOFTWARE FLOPPY LABELS>>				
VIDEODISC LABELS>>				
3 RING BINDERS>>				0
3 RING BINDER TITLE PAGE>>				0
3 RING BINDER TABS>>				0
USER'S GUIDES>>				0
COORDINATOR'S GUIDES>>				

PRODUCTION REQUIREMENTS (total for all sides)

	number	0 to 10 production value	0 to 100 % complete	Amt.
SETS>>				0
SPECIAL PROPS>>				0
SPECIAL COSTUMES>>				0
PREPRODUCTION DAYS>>				0
PRODUCTION DAYS>>				0
WRAP DAYS>>				0
DAYS - ADDITIONAL AUDIO>>				0
OFF-LINE EDITING DAYS>>				0
AUDIO EDITING DAYS>>				0
ON-LINE EDITING DAYS>>				0
SWEETENING DAYS>>				0
DAYS - MAJOR ACTORS (VIDEO)>>				0
DAYS - MAJOR ACTORS (AUDIO)>>				0
DAYS - MINOR ACTORS (VIDEO)>>				0
DAYS - MINOR ACTORS (AUDIO)>>				0
MINUTES - ORIGINAL MUSIC>>				0
NEEDLEDROPS>>				0

Evaluation

	number of disk sides	Amt.
SYSTEM TESTING>	0	3400
ALPHA TESTING>>		400

Project Requirements Document

MARKETING

	yes/no	1 to 10 degree of complexity	0 to 100 % complete	Amt.
BUSINESS PLAN>>				
MARKETING STUDY>>				

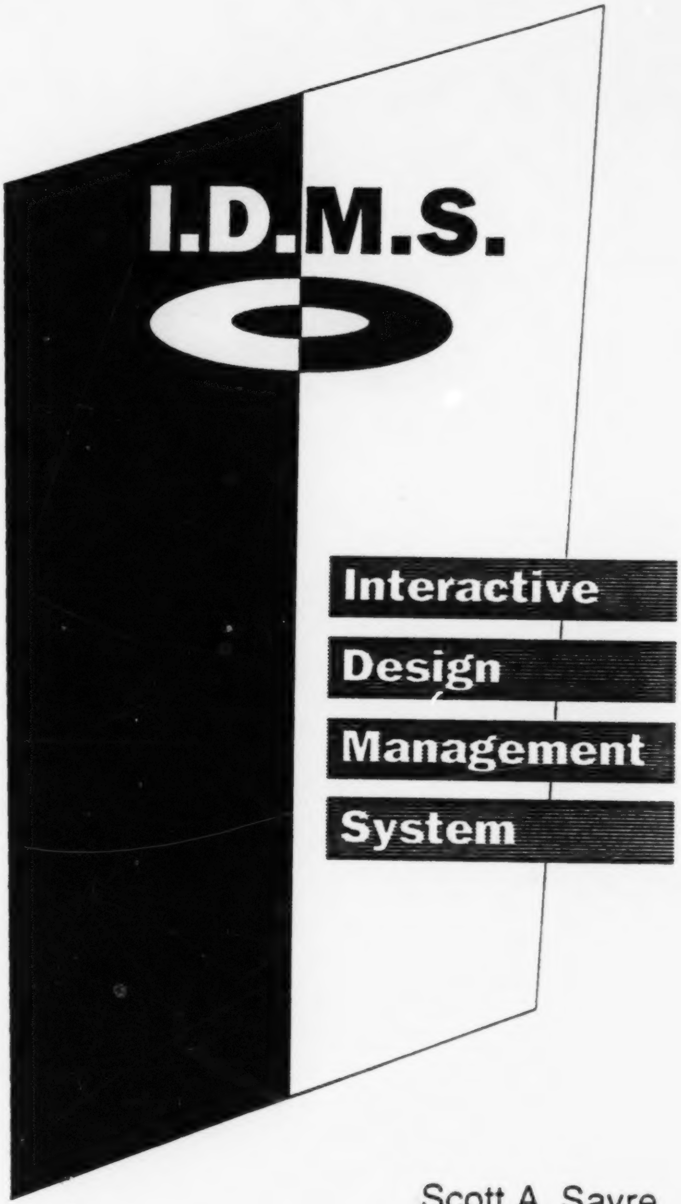
CONTINGENCY FACTORS

	$
ENTER A PERCENTAGE TO BE APPLIED TO THE SUB-TOTAL DOLLAR AMOUNT FOR ALL THE ABOVE "PRODUCTION REQUIREMENTS">>	

COMMENTS

Appendix D

IDMS Software Manual



I.D.M.S.

Interactive
Design
Management
System

Scott A. Sayre

Interactive Design Management System

by Scott Sayre

© Synapse Research
89 North Victoria Street
St. Paul, Minnesota 55104
(612) 290-0042

Interactive Design Management System
Synapse Development- January 1990

Software and manual design and development
by Scott Sayre

Technical documentation
by Rose Ripka and Scott Sayre

Desktop publishing
by Merri Fromm and Scott Sayre

Software developed with
Silicon Beach's SuperCard

Documentation created with
Aldus PageMaker

All Rights Reserved
Copyright © by Scott Sayre

Contents

Foreword vii
How this manual is organized viii

Chapter 1. An introduction to IDMS

What is IDMS ? 1
Who was IDMS designed for ? 1
What's special about IDMS ? 1
What tools does IDMS provide ? 2
What do I start with ? 3
What do I end up with ? 3

Chapter 2. Getting started with the IDMS system

Computer setup 5
Starting a new project 6
IDMS menus 7
IDMS terminology 10
Other important considerations 12
Where do I go from here ? 13

Chapter 3. Analysis Outliner

Using the Analysis Outliner 15
Opening the Analysis Outliner 16
The Outliner menu 17
The Font menu 17
Using Analysis Ideas 17
Using the Outline Writer 18

Chapter 4. Intelligent Project Schedules

Using Project Schedules 21
The Elements menu 22
The Page menu 23
The schedule pages 24
Preparing the schedule page 26
Calculating schedule status 26
The staff responsible list 26
Searching staff responsibilities 27
The task linking option 28
Printing schedules 29

Chapter 5. Flowchart Processor

Introduction to the Flowchart Processor 31
Th Flowcharts menu 32
The Icons menu 3
The flowchart tools palette 33
Adding a new icon 33
Linking an icon to a template page 34
Moving existing icons 35
Autoscrolling the browse tool 35
Template/Flowchart Links 36
Creating a new chart page 36
Changing the chart size 36
Changing chart background pattern 37
Navigating among chart pages 37
The icon connecting tool 38
Send to back & bring to front commands 38

Chapter 6. Interactive Design Templates

Using the Design Tempalates 39
The Templates menu 42
The Find/Sort menu 43
Sections of the Template page 43
Page information section 44
Using the Template pages 49
Template branching 52
Interaction with the Flowchart Processor 54
Using the Template Libraries 56
Navigating through the template pages 57
Printing Template pages 61

Chapter 7. The Accessories

Using the Accessories 63
The Accessories menu 63
Using the Program Launcher 64
Using Sticky Notes 66
Using the Analysis Ideas 67
Using the Outline Writer 68
Using the Frame Calculator 71
Using the Script Calculator 73

Chapter 8. Developing an IDMS work procedure

What is the best work procedure? 75
Assorted tips and techniques 75
An outline of one possible work procedure 76

Appendix I. The IDMS process model 79

Appendix II. Blank Forms 85

Index 93

Note: This page has been intentionally left blank to maintain the indexing of the original software manual within this thesis appendix.

Foreword

The Program's Roots

After my first encounter with interactive videodisc (IV) in 1983, I, like many people, was fascinated. As a photographer, printer, animator, audiophile, video producer, amateur programmer, and educator, I was excited by the potential of this medium. However, after investigating all of the variables IV program designers face, I saw that the task of designing IV programs could be quite overwhelming. I kept asking myself how IV designers and producers could accurately previsualize and test the effect and flow of a truly interactive program prior to production. Paper and pencil methods weren't the answer.

In 1987, Bill Atkinson's revolutionary Hypercard inspired me to consider the use of integrated data, graphics, audio and presentation software in the IV program design process. Of course, Hypercard has its limits and so did the pilot stacks I experimented with for the next two years. It wasn't until the release of Silicon Beach's SuperCard in 1989 that my integrated approach could be fully developed and tested.

The resulting Interactive Design Management System is one of the products of my Doctoral thesis, "The Development of a Computer-based System for Managing the Design and Pilot-testing of Interactive Videodisc Programs" at the University of Minnesota. The eleven-element interactive videodisc design model contained within the system was developed with information collected from Applied Learning, Arthur Andersen Consulting, Minnesota Extension Service, Northwest Aerospace Training, Emerging Technology Consultants, and Wilson Learning. Additional information was gathered through a review of current literature. Support for the development of this product was provided by Project MinneMac (a cooperative grant program between the University of Minnesota and Apple Computer, Inc.) and the University of Minnesota's Training and Development Research Center. I am very grateful to all of these organizations.

As a noncompiled, "script-based" program, IDMS is only as powerful as the system which supports it. I sincerely hope someday to expand on these design system concepts without the constraints imposed by an "authoring system." With that in mind, I welcome your suggestions, comments, ideas, and opportunities.

I hope you enjoy IDMS,

How this manual is organized

Style:	Indicates:
Underline -	a menu name
Bold -	a menu item
Outlined -	a button name
Shadowed -	a tool name
User /Computer	Interaction:
Numbered tasks-	user performed
Bulleted tasks (•)-	computer performed

CHAPTER 1
An introduction to IDMS

After reading this chapter you will:

—understand the purpose of IDMS
—understand who can use and benefit from IDMS
—understand the types of tools IDMS provides
—understand what you need to start an IDMS project
—understand the end product of an IDMS project

What is IDMS ?

The Interactive Design Management System (IDMS) is an integrated solution to the problem of managing the large amount of complex information required in interactive videodisc (IV) program design. It offers a hypermedia working environment for designers, developers, and producers of interactive media programs.

Who was IDMS designed for ?

IDMS was designed to fit the needs of all people interested in using the computer for designing and managing IV projects. Even those who choose to work on paper may use IDMS as a system for generating blank design templates and schedules. Instructors teaching IV development can also use IDMS as a unique method for allowing students to design and test IV programs with no production costs.

What's special about IDMS ?

Interactive videodisc programs consist of an integration of theories and technologies from a variety of different disciplines. The successful design and production of these "hybrid" programs is dependent upon the project manger's, designer's, and producer's ability to consider and control all aspects of the program's design and production. Software-based tools such as word processors, databases, and graphic programs are often used to produce and manage project development information. However, these tools are limited by their "general purpose" design and independence from one another.

IDMS gives you the opportunity to combine the capabilities of many of the software applications commonly used in IV program development within an integrated system. This integrated approach allows software tools to be linked together, share information, and even simulate program interactivity. IDMS offers you the interactive design management tools you need for IV program development.

What tools does IDMS provide ?

IDMS provides IV program designers and project managers with a unique set of integrated software tools called "Hypertools." The Hypertools include **"Intelligent" Project Schedules, an Analysis Outliner, a Flowchart Processor, and Interactive Design Templates**. These Hypertools can be used independently or connected in a variety of combinations, depending on the needs of each specific project.

"Intelligent" Project Schedules are a series of project management matrices which allow you to monitor the current status of each task in the design process at a glance. The **Analysis Outliner** consists of a simple word processor and a computer-based notebook containing a series of instructional design and project management questions to aid in the IV program development process. The **Flowchart Processor** provides tools for constructing flowcharts to help visualize the flow of the program. Flowcharts can be linked to the **Interactive Design Templates** which are the most powerful component of IDMS. Each template page serves as an interactive form for recording the details of each program sequence. Branches can be created between template pages, allowing you to pilot-test the user interactivity and program flow while the program is still in the design phase.

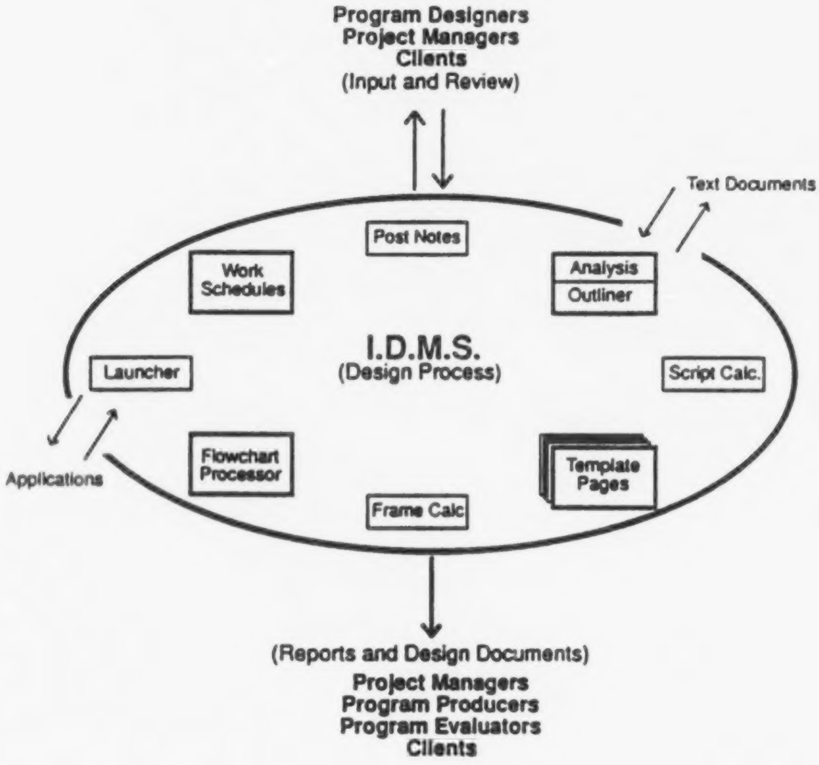
In addition to the Hypertools, IDMS provides a set of powerful Accessories similar to the Macintosh's Desk Accessories. These Accessories are available at all times in IDMS and were designed specifically to aid in the IV design process. The Accessories include: a **Program Launcher** for launching external applications; **Sticky Notes** for writing and pasting Post-It™ type notes; the **Analysis Ideas** notebook containing chapters of IV related questions; the **Outline Writer** for creating,, importing, and exporting text documents; **Frame Calculators** for converting time code values into video frames; and a **Script Calculator** for estimating narrative script durations.

What do I start with ?

All you need to start an IDMS project is an instructional problem in need of a solution. Even if IV does not turn out to be the answer, the Analysis Outliner will help you define your problem, consider your alternatives, focus your thoughts, and begin working towards a solution. Project Schedules can be used to organize your time and set deadlines. As the project grows, the other Hypertools and Accessories can be incorporated if you find the IV or computer-based instruction is part of the solution.

What do I end up with ?

The products of IDMS serve a dual purpose. During the initial design process, the documents produced with IDMS can be used as references for maintaining design consistency and locating content areas requiring further work. Once the content of a program has been developed and the branching has been tested pilot-tested, the IDMS documents can be printed to produce a fully detailed design document to be used in production of the program.


Program Designers
Project Managers
Clients
(Input and Review)
Text Documents
Post Notes
Work Schedules
Analysis
Outliner
I.D.M.S.
(Design Process)
Launcher
Script Calc.
Flowchart Processor
Template Pages
Frame Calc
Applications
(Reports and Design Documents)
Project Managers
Program Producers
Program Evaluators
Clients

Note: This page has been intentionally left blank to maintain the indexing of the original software manual within this thesis appendix.

File
Edit
Go
Hypertools
Accessories

CHAPTER 2
Getting started with the IDMS system

After reading this chapter you will be able to:

—understand IDMS's hardware requirements
—install IDMS on your computer's hard disk
—open IDMS
—set IDMS project preferences
—use the standard IDMS menus
—understand IDMS terminology
—decide where to start

Computer Setup

Hardware Requirements

Macintosh SE with hard disk for small projects
Mac II with hard disk for medium to large projects
1MB RAM (minimum)
2+ MB RAM (recommended)

Installing IDMS On Your Hard Disk

The operation of IDMS is unique from most software programs. It is a project dedicated system. Once a project is started, the system customizes itself to work specifically on that project. Managing several projects with IDMS requires separate sets of IDMS files and folders for each project. Each set of project files must reside in a separate folder to avoid errors and unwanted data transfers. This type of dedicated system approach allows unusual degree of software flexibility.

> Warning: Because of this system's dedicated design, it is important that you do not attempt to run the IDMS files on the master diskettes. Always copy the IDMS files onto your computers hard disk before attempting to run the system.

Your IDMS master diskettes should contain the following files:

1) IDMS (application)
2) Schedules (file)
3) Analysis Outliner (file)
4) Flowchart Processor (file)
5) Template Library 1 (file)
6) Print (file)

Each time you want to create a new IDMS project file the following steps must be completed...

1. Estimate the amount of free disk space on the computer's hard disk. You will need approximately 1MB to install IDMS. Once you begin designing, IDMS's files will continue to grow much larger. Small projects consisting of less than 50 template pages will probably remain under 2 Mb in size. However, large projects may end up exceeding 5 Mb. It is important that you check your available disk space each time before you start IDMS to avoid unexpected disk space problems.

2. Create a new folder and give it a unique name (preferably that of the new project).

3. Copy all IDMS files from both IDMS master floppy disks onto the computer's hard disk.

   - IDMS is now operational!

Starting a New Project

Opening IDMS

The procedure for opening (running) IDMS is similar to that of other Macintosh applications.

1. Open the IDMS application by double clicking the IDMS application icon or name in the finder.

   - the IDMS introductory screen will appear

   - the IDMS system menus will appear

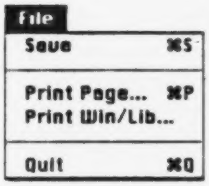
Setting Project Preferences

This option sets defaults for document headings and labels your project. Once a project name is entered, it stays on the system for the remainder of the project. The Project Prefs. only have to be set once, when opening a new project.

1. Select **Project Prefs.** from the Hypertools menu.
   - the project preferences dialog box will open
2. Enter the program title, estimated starting date, estimated completion date, and the project director's name.
3. Click the **OK** button to save the entries.

IDMS menus

There are standard pull-down menus that will be available regardless of what component you are using. These standard menus include: File, Edit, Go, Hypertools, and Accessories. Other menus will appear, depending upon the Hypertools and accessories which are currently in use.

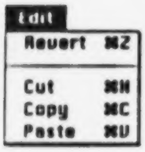


The File Menu

The File menu allows you to save or print file information.

Save	saves all changes in the top window since the last page change or user-initiated save. IDMS automatically saves all changes each time you quit the program. Frequent saving also makes processing more efficient.
Print Page...(⌘P)	prints the entire page in the topmost window, regardless of the top window size
Print Win/Lib...	prints all pages included in the topmost window or template library. Prints all the Outliner files when used with the Outline Writer.

Quit (⌘Q)	closes the IDMS program and saves all changes made since the last user-initiated or automated save was executed

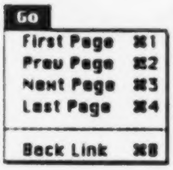


The Edit Menu

The Edit menu provides text, graphic, and page editing commands.

Revert (⌘Z)	is similar to the Macintosh Undo command; will undo all changes made since the last user-initiated or automated save was executed.

Cut (⌘X)	lets you cut selected graphics or text from its current location and places a copy of it on the clipboard
Copy (⌘C)	lets you make a copy of selected text or graphics and places a copy of it on the clipboard
Paste (⌘V)	allows you to paste a copy of what you have cut or copied onto the clipboard wherever you place the cursor. Text placed in the memory of the clipboard can be pasted more than one time. It remains in the clipboard memory until you copy something else or you quit.



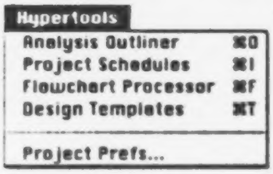
The Go Menu

The Go menu can be used to move between pages (design templates, flowcharts, schedules) residing in the currently open window. For example, you "flip" through the all pages in a template library (window) using the Next Page command. Next Page and Previous Page commands can also be sent using the right and left keyboard arrow keys.

First Page (⌘1)	takes you to the first page of the current window
Prev Page (⌘2)	takes you to the previous page of the current window
Next Page (⌘3)	takes you to the next page of the current window

Last Page (⌘4)	takes you to the last page of the current window

Back Link (⌘B)	a special command for returning to the project schedules after viewing a document linked to a task



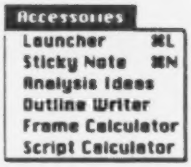
The Hypertools Menu

The Hypertools can be used interactively. If you want to open a specific Hypertool, select it on the Hypertools menu.

If you are already working in a document and have a window open, you can select another Hypertool and IDMS will bring that tool-specific window to the current (top) position.

Project Schedules (⌘I)	opens the project schedule window and inserts the Schedules menu
Analysis Outliner (⌘O)	opens the Outline Writer, the Analysis Ideas notebook, and inserts the Outliner menu
Flowchart Process.(⌘F)	opens the flowchart processor window, and inserts both the Flowcharts menu and the Icons menu
Design Templates (⌘T)	if a template library is open, a dialog box appears and asks you to select the template library you want to open; it then opens the design templates window, and inserts both the Templates menu and the Find/Sort menu.

Project Prefs....	opens a dialog box which allows you to specify your project's name, starting date, and ending date. **Project Prefs....** then prints your selected name and dates on all document headers that are part of this particular project. **Project Prefs....** is not a Hypertool — it is an initial step in the IDMS process designed for program efficiency.



The Accessories Menu

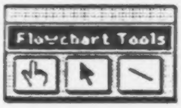
Accessories perform specific functions often required in the design process. You can use the Accessories menu to calculate duration times or the number of video frames, add text or audio scripts, make notes to yourself anywhere within any document, or a variety of other functions. See Chapter 7 for more information on using Accessories.

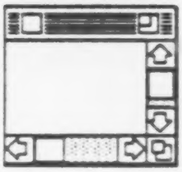
Program Launcher (⌘L)	allows you to launch external applications without having to return to the finder. You can customize the launcher to launch up to twelve different applications.
Sticky Notes (⌘N)	creates and paste a Post It™ type note anywhere in the IDMS system
Analysis Ideas	opens a computer-based notebook containing a series of questions and ideas related to interactive videodisc design and project management.
Outline Writer	opens a simple word processor for the editing, calculating, cutting and pasting of narrative scripts. The editor is capable of exporting and importing text documents to and from other applications.
Frame Calculator	opens a special calculator for calculating durations from time code values and converting durations into video frames
Script Calculator	opens a special calculator for estimating script duration times.

IDMS terminology

IDMS was developed using SuperCard™, a software construction tool very similar to Apple's Hypercard. User's familiar with Hypercard's interface and terminology should find many similarities in IDMS. However, because SuperCard™ offers some additional, non-Hypercard features, and not all users are familiar with Hypercard some definition of terminology is necessary

☐	Box	a small square box similar to Macintosh's checkbox which automatically marks itself when clicked
☝	Browse tool	a hand-shaped cursor which is used to click on buttons and boxes
OK	Buttons	geometric shapes which perform special functions when clicked. Buttons are often in the shape of ovals, rectangles,and arrows
⌖	Connector Tool	a circle and crosshair shaped cursor used to draw connecting lines in the Flowchart Processor
	Clicking	the process of moving the cursor over an object such as a button or box and pressing the mouse button. Most IDMS functions only require a single mouse click
	Current window	the window which is on top of all other open windows, not including palettes
	Dialog Box	a large rectangular box with double borders used to ask the user for information
	Double-click	the process of rapidly clicking twice on a screen object. IDMS requires double clicking only when moving flowchart icons beyond the borders of the current window or selecting dialog menu items
	Fields	rectangular areas on IDMS documents for entering text and numeric data
	Icons	special buttons used in the Flowchart Processor to represent program sequences
➤	Icon Mover Tool	a arrow shaped cursor used in the Flowchart Processor to move flowchart icons
	Page	a single IDMS document (outline file, schedule page, flowchart page, template) residing within a Hypertool window





Palette	a special kind of window which floats above the Hypertool windows. IDMS uses palettes for such things as page linking, tool selection and carrying accessories
Window	a rectangular area with scroll bars on the right and bottom edges. IDMS uses windows to contain Hypertool documents.

Other important considerations

Saving Files

Similar to Hypercard, whenever you close a document or go to a new document page in IDMS, the changes are saved automatically.

Window Management

IDMS allows you to have have multiple windows open simultainiously. However, the system runs most efficiently when a minimum number of windows are open. Open windows take up memory space and slow processing time, so try to open only the windows that you need.

If you want to refer to another Hypertool window simultaneously, you can open the window and resize both windows into a workable size. Reducing window size also reduces the processing time.

Operating Speed

IDMS was written in SuperTalk™, an uncompiled scripting language. Because of this, IDMS runs noticeably slower than common compiled applications. If used efficiently IDMS will save you time, however you will find that certain tasks, such as updating schedules, finding, and sorting, will take a considerable amount of time. Just remember, the computer's doing the work... Take a moment to enjoy your new found freedom.

Use of Consistent Terminology

It's good idea to develop a set of standards for project related terminology at the onset a project. This will assist you and the others involved in the project in accurately finding and sorting recorded project information as the project's design matures. Abbreviations and acronyms require particular attention since they are often unconsciously and randomly used.

Where Do I Go From Here ?

Now that you have IDMS installed and have a basic understanding of system, you're ready to begin a project. Chapters 3 through 7 provide detailed descriptions of the purpose and operation of each of the Hypertools and Accessories. You may choose to use these remaining chapters in a number of different ways depending upon your learning/working style.

"I want to use it now!"

If you're an intuitive Macintosh user you may just want to open a Hypertool and begin working, using the rest of the manual as a reference. The index in the back of the manual will help you find your answers fast.

"Too many options... Where do I start ?"

Because IDMS does not force you to work in a specific way, you may find it difficult to jump right in and start. One way to sort out the options and get started is to read the summary's at the beginning of each of the following chapters. These will help you get a better understanding of the applications and options of each of the Hypertools and accessories. You can then begin a project by following the example described in chapter 8 and reference the unread sections of the manual when necessary. This way you'll be able to modify the example to fit you own project needs.

"I want to know exactly what I'm getting into..."

Lots of people prefer to know all their options right from the start. If your nervous about what your getting into or want to harness all the power of IDMS from the beginning, read chapters 3 through 7. You'll then be ready to start off on your own, or run through the example in chapter 8.

No matter where you begin...

After you begin working with IDMS, you'll establish your own work procedures and transform IDMS into your "own" system. Enjoy the new IV design options IDMS provides...

Note: This page has been intentionally left blank to maintain the indexing of the original software manual within this thesis appendix.

File
Edit
Go
Hypertools
Outliner
Font
Accessories

CHAPTER 3
Analysis Outliner

After reading this chapter you will be able to:

—understand the purpose and capabilities of the Analysis Outliner
—use the Analysis Ideas notebook as a reference tool
—create text documents with the Outline Writer
—import and export text documents with the Outline Writer
—print Outline Writer documents

Using the Analysis Outliner

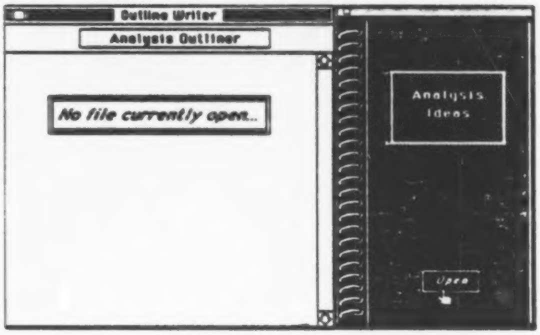
The analysis outliner is a brainstorming tool for both encouraging and recording project related thoughts. It consists of two components: the Analysis Ideas Notebook and the Outline Writer. These components may be used together, as the Analysis Outliner, or separately as accessories to other hypertools.

Analysis Ideas

This component is a computer-based notebook containing a series of questions and ideas related to interactive videodisc design and project management. The notebook consists of eleven chapters, each representing an element of the IDMS model (see appendix 1). Each chapter contains questions related to the suggested tasks contained in an IDMS element. Like the tasks within the IDMS model, these quetions are neither exhaustive nor required. They are to be used only as fuel for the design and management process.

Outline Writer

The Outline Writer is a simplified word processor for writing and saving information. Although the Analysis Outliner's primary purpose is to record responses to Analysis Ideas, it may also be used independently for editing scripts and writing notes. It's ability to import and export text documents increases IDMS's compatibility with other text-based applications.



No file currently open...

Opening the Analysis Outliner

As a Hypertool

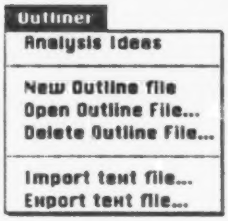
1. Select **Analysis Outliner** from the Hypertools menu.
   - the Outliner menu will appear
   - the Outline Writer window will open
   - the Analysis Ideas notebook will open

As Separate Accessories

1. Select **Analysis Ideas** from the Accessories menu.
   - the Analysis Ideas notebook will open

or

2. Select **Outline Writer** from the Accessories menu.
   - the Outliner menu will appear
   - the Outline Writer window will open


Outliner
Analysis Ideas
New Outline file
Open Outline File...
Delete Outline File...
Import text file...
Export text file...

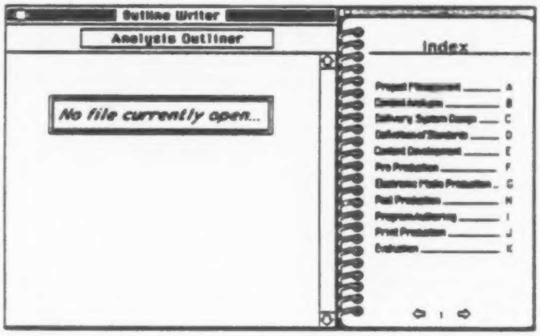
The Outliner Menu

The Outliner menu appears automatically when **Analysis Outliner** (I) is selected under the Hypertools menu or **Outline Processor** is selected under the Accessories menu.

Analysis Ideas	opens and closes the Analysis Ideas notebook
New Outline File...	creates a new Outliner file
Open Outline File...	opens an existing outliner file
Delete Outline File...	deletes an existing outliner file
Import Text File...	creates an outliner file from an external text file
Export Text File...	creates an external text file of the currently open outliner file

The Font Menu

The Font menu automatically lists all of the fonts installed in the computer's system file.

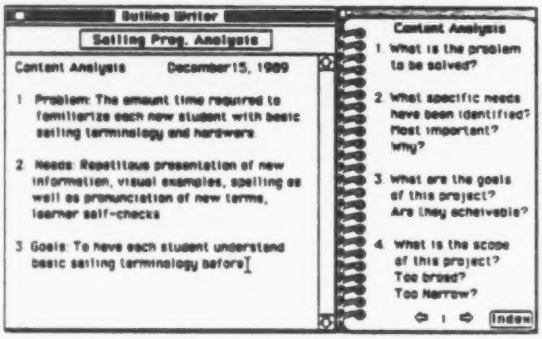

Outline Writer
Analysis Outliner
No file currently open...
Index
1

Using Analysis Ideas

1. Open the Analysis Ideas notebook as either a Hypertool or an Accessory (refer to Opening the Analysis Outliner above).
2. Click the **Open** button on the notebooks cover.
   - the contents page will appear

3. Click on the chapter's title you wish to view
   - the specified chapter will open
4. Use the **<-** and **->** buttons at the buttom of the pages to "browse" forward and backward through the chapter's pages.
5. Click the **Index** button on the bottom of the page to return to the table of contents page.

Using the Outline Writer



Content Analysis December 15, 1989

1 Problem: The amount time required to familiarize each new student with basic sailing terminology and hardware.

2 Needs: Repetitous presentation of new information, visual examples, spelling as well as pronunciation of new terms, learner self-checks

3 Goals: To have each student understand basic sailing terminology before

Content Analysis

1. What is the problem to be solved?
2. What specific needs have been identified? Most important? Why?
3. What are the goals of this project? Are they acheiveable?
4. What is the scope of this project? Too broad? Too Narrow?

Opening the Outline Writer

Open the Outline Writer as either a Hypertool or Accessory (refer to Opening the Analysis Outliner above).

Creating a New Outliner File

1. Select **New Outline File...** from the Outliner menu.
   - a dialog box will open asking you to name the new file
2. Enter a file name and click the **OK** button in the dialog box.
   - a new Outline Writer page will appear with the file name displayed in the header
3. Type your document in the Outline Writer window.
   - all entries and changes will automatically be

saved when the Outline Writer window is closed or another outline file is opened

Opening an Existing Outliner File

1. Select **Open Outline File...** from the Outliner menu

   • a dialog box will open asking you to select the file you want to open

2. Double click on the file name **or** select it and click the **Open** button to open the file

   • the selected file will appear in the Outline Writer window

Deleting an Outliner File

1. Open the outliner file you want to delete
2. Select **Delete Outline File...** from the Outliner menu

   • a dialog box will open asking you if you want to delete the file

3. Click the **OK** button to delete the file or the **Cancel** button to abort the delete.

Importing a Text File

1. Select **Import Text File** from the Outliner menu.

   • a dialog box will open asking you to select the file you want to import

2. Double click on the name of file you want to open **or** select the file's name and click the **Open** button.

   • a copy of the file will open in the outline writer window and the name of the file will be displayed in the heading. The original text file will still exist outside IDMS.

Exporting a Text File

1. Open the outline file you wish to export.

2. Select **Export Text File...** from the Outliner menu.

   - a dialog box will open asking you to name the new text file

3. Rename the file or leave the name the same as the Outliner file.

   Note: Since text files reside outside of the IDMS system, they may have the same name as outliner files.

4. Click the **Save** button.

   - a text file is created in the IDMS folder

Changing the Font

The Font menu allows you to change the font of any selected text in the Outline Writer window. These font changes only effect the text displayed on the screen and will not have any effect on exported text files or files printed from within IDMS.

1. Select the text you want the font change to effect.
2. Select the new font from the Font menu.

   - the font of the selected text will be changed

Printing Outliner Files

In most cases it is recommended that you export, edit and print important outliner files with a standard word processor since it will allow you greater formatting capabilities. However, you can print a hard copy of any or all of the Outliner documents using the **Print Page** and **Print Win/Lib** commands.

To print a single a outliner file:

1. Open the Outline Writer file you wish to print.
2. Select **Print Page** from the File menu.

   - the print dialog box will open

To print all outliner files:

1. Open the Outline Writer.
2. Select **Print Win/Lib** from the File menu.

   - the print dialog box will open

CHAPTER 4
Intelligent Project Schedules

After reading this chapter you will be able to:

—understand the purpose and capabilities of the Project Schedules
—understand the relationshipr between the Project Schedules
—and the IDMS model
—customize schedule pages to best fit the needs of each project
—calculate and interpret the task status indicators
—link schdule tasks to other IDMS documents
—use the staff responsible funtion to identify an individual staff menbers responsibilities
—print schedule pages

Using Project Schedules

The "intelligent" project schedules allow you to visually monitor the current status of each task in the design process. The schedules contain the eleven element IDMS model which lists over 180 tasks typically involved in videodisc program design and production. You can customize these schedules to fit your project's needs by simply adding or deleting tasks.

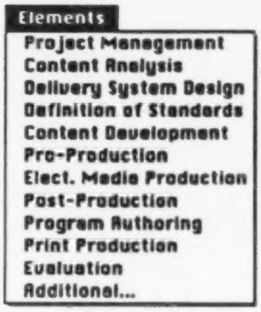
A searchable, staff responsible list allows you to assign tasks and then quickly monitor each individual's responsibilities. The "task linking" option links specific tasks to other documents within IDMS so that related documentation can be viewed with the click of a button. Any or all of the schedule pages may be printed for distribution or record keeping.

The IDMS Model

The IDMS model contains eleven interactive videodisc development elements. Each element is represented on its own schedule page. Each schedule page contains a list of recommended design tasks related to the specific part of the design process. The tasks listed are neither exhaustive nor required — they are provided only as a guide and may be edited to fit each specific project.

Opening the Project Schedules

1. Select **Project Schedules** (I) from the Hypertools menu.
    - the Elements an Page menus will appear
2. Select the schedule page you want to view under the Elements menu
    - the corresponding schedule page will open

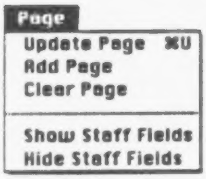

Elements
Project Management
Content Analysis
Delivery System Design
Definition of Standards
Content Development
Pre-Production
Elect. Media Production
Post-Production
Program Authoring
Print Production
Evaluation
Additional...

The Elements Menu

The following is a list and description of the elements of the IDMS model found under the Elements menu. By selecting an element menu item, you open a representative schedule page. Refer to Appendix C for a complete listing of the tasks included under each element.

Project Management	organizing, monitoring, and controlling all project related tasks and resources
Content Analysis	specifying a program's conceptual and physical parameters based on environmental needs and internal constraints
Delivery System Design	specifying the configuration and capabilities of the authoring environment and delivery hardware
Definition of Standards	constructing an overall set of program design rules and parameters to which all program content must adhere
Content Development	identifying, collecting, organizing and refining information into a workable instructional design

Pre-Production	identifying and controlling all production components prior to the production process; generating all documentation necessary to accurately communicate the instructional design and production requirements to the media producers and talent
Elect. Media Production	creating and recording all necessary audio and visual materials
Post-Production	organizing and integrating all recorded materials into a program according to a preconceived design
Program Authoring	developing a software-based program which controls an instructional program's delivery and interaction
Print Production	identifying, designing, and producing all printed information
Evaluation	constructing strategies and instruments for validating a finished program



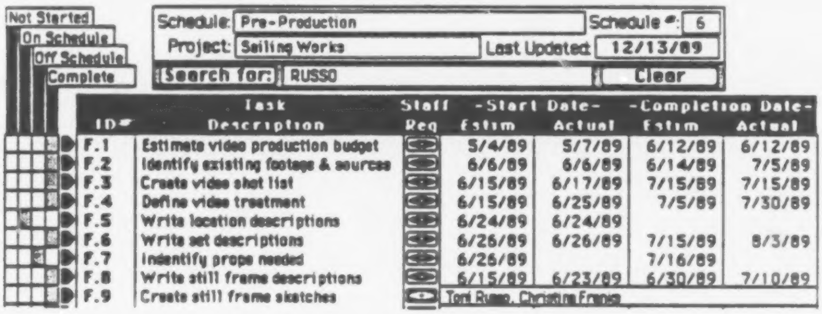
The Page Menu

The Page menu appears automatically when Project Schedules (I) is selected from the Hypertools menu.

Update Page (U)	scans estimated start and completion dates and visually depicts the status of the tasks in the status blocks located on the far left of the schedule.
Add Page	creates a new blank schedule page after the last schedule page (highest page number)
Clear Page	erases all data from the page except for the task ID numbers and the task descriptions

Show Staff Fields	displays the staff responsible fields for all tasks on that page
Hide Staff Fields	hides the staff responsibilities fields for all tasks on that page

The Schedule Pages



Schedule: Pre-Production
Schedule #: 6
Project: Sailing Works
Last Updated: 12/13/89
Search for: RUSSO

ID#	Task Description	Staff Req	-Start Date- Estim	-Start Date- Actual	-Completion Date- Estim	-Completion Date- Actual
F.1	Estimate video production budget		5/4/89	5/7/89	6/12/89	6/12/89
F.2	Identify existing footage & sources		6/6/89	6/6/89	6/14/89	7/5/89
F.3	Create video shot list		6/15/89	6/17/89	7/15/89	7/15/89
F.4	Define video treatment		6/15/89	6/25/89	7/5/89	7/30/89
F.5	Write location descriptions		6/24/89	6/24/89		
F.6	Write set descriptions		6/26/89	6/26/89	7/15/89	8/3/89
F.7	Indentify props needed		6/26/89		7/16/89	
F.8	Write still frame descriptions		6/15/89	6/23/89	6/30/89	7/10/89
F.9	Create still frame sketches		Toni Russo, Christina Francis			

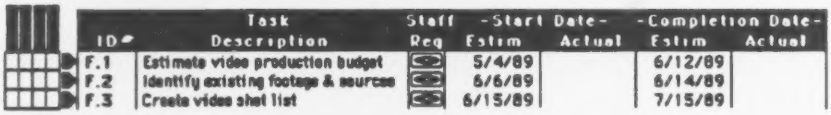
The following is a list of the parts of the schedule page and a description of their uses.

Schedule Name (field)	contains the name of the program development element this page represents
Schedule Page # (field)	contains the number of the schedule page
Last Updated (field)	contains the date on which the schedule was last updated
Search For (button)	searches the Staff Responsible fields for the name entered in the Search For field
Search For (field)	for entering the staff member's name you want to search for
Clear (button)	clears all the highlighted Staff Responsible buttons after a staff responsible search

Status Block Columns	indicates the current status of each task by darkening the appropriate blocks. These blocks become active when an estimated start date is specified for a task. If there are no tasks assigned, then the blocks are empty.
Task Linkers (buttons)	can be used to link a task to a specific document in the IDMS system. If the Linking Button is a solid black arrow, it is not linking a document to a task. When the arrow is outlined in black, it is linking a task and a document.
Task ID Number Column	allows you to assign up to a four-digit number to each task
Task Description Column	allows you to describe each specific task.
Staff Responsible (buttons)	show and hide the staff responsible fields
Estimated Start Date Column	allows you enter the estimated starting date for each task*
Actual Start Date Column	allows you to enter the actual date each task is started on*
Est. Completion Date Column	allows you enter the estimated completion date for each task*
Act. Completion Date Column	allows you to enter the date on which the task was actually completed*

* The dates must be entered in the following format: 12/3/89.

Preparing Schedule Pages



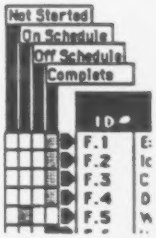
ID	Task Description	Staff Req	-Start Date- Estim	Actual	-Completion Date- Estim	Actual
F.1	Estimate video production budget		5/4/89		6/12/89	
F.2	Identify existing footage & sources		6/6/89		6/14/89	
F.3	Create video shot list		6/15/89		7/15/89	

Input of Work Dates

The accuracy of the status block indicators is dependent upon the information entered in the schedules columns. In order for a task's status blocks to become active, a date must be entered in the estimated start date column. Each time dates are entered or changed, the schedule should be recalculated. The dates must be entered in the following format: 4/1/90.

Editing Schedules

You can customize the schedules to fit your specific needs. Text can be entered, deleted, cut, copied, and pasted in any of the fields on a schedule page. Menu commands, from the Edit menu, and keyboard editing commands can be used to increase the efficiency of the editing process.

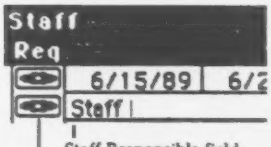


Calculating Schedule Status

To update a schedule page's status block to represent the current date or date changes, select **Update Page** (U) from the Page menu.

The Staff Responsible List

Each task is connected to a separate staff responsible field. Staff responsible fields are used to list the staff members assigned to each of the tasks.



Staff Responsible field

Staff Responsible button

Assigning Staff Responsibilities

Click on the representative **Staff Responsible** button.

- a staff responsible field will appear for you to enter the name(s) of the responsible staff member(s)

or

Select **Show Staff Fields** from the Page menu to display all staff fields.

- all staff responsible fields will appear for you to enter the name(s) of the responsible staff member(s)

Note: To simplify future searching, it is recommended that you standardize how you enter names (eg., T. Edison).

Hiding Staff Responsible Fields

Press on the representative **Staff Responsible** button to hide an individual staff field.

- the corresponding responsible staff field will disappear

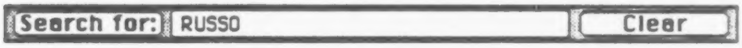
or

Select **Hide Staff Fields** from the Page menu after you have viewed or entered information to hide all staff fields.

- all responsible staff fields will disappear

Searching Staff Responsibilities

This function finds each instance in which the specified name occurs in the responsibility fields of the currently open schedule page.



Search for: RUSSO

Performing a Staff Search

1. Type the desired staff member's name into the **Search For** field.

2. Click on the **Search For** button.

   - The **Staff Responsible** buttons will be highlighted next to each task that particular person is responsible for.

3. Click the **Staff Responsible** buttons to view the responsibilities field and the other staff members responsible for that task.

4. To perform a multiple name search, repeat steps 1-3 without clearing previous searches.

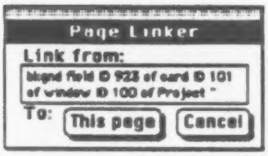
Clearing a Staff Search

1. Click on the **Clear** button.

- the currently highlighted **Staff Responsible** buttons will be cleared

The Task Linking Option

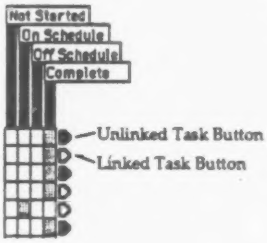
Task linking allows you to link each task to a related document (eg. an Outliner file) contained in the IDMS system. Once a link has been established, the related document may be accessed with the click of the associated **Task Linker** button.



Link from: bkgnd field ID 923 of card ID 101 of window ID 100 of Project "

To:

Creating a Task Link



1. Click on a **Task Linker** button associated with a specific task line. The **Task Linker** button is a solid black arrow.
   - a page linker palette will appear
   - the schedule page will close

2. Open the specific page of any IDMS document you wish the task to be linked to.

3. Click on the **This Page** button on the page linker palette.
   - the task link has been made
   - the original schedule page reappears and the **Task Linker** button appears outlined in black

Traveling to a Linked Document

1. Click on the representative **Task Linker** button.
    - the window of the linked document will then appear as the current window
2. Select **Back Linker** from the Go Menu to return to the schedule page.

Deleting a Task Link

Hold down the command key and simultaneously click on the outlined **Task Linker** button you wish to delete.

- the **Task Linker** button will return to its solid status

Viewing a Task Link Path

At times, it may be useful to check what document a task is linked to without traveling to the document. This can be done by viewing the link path.

1. Hold down the command key ⌘ and simultaneously click on a linked button.
    - the field appears describing the path to the linked document
2. Repeat step 1 to hide the path field.

Printing Schedules

A Single Schedule

1. Open the schedule page you want to print.
2. Select **Print Page...** from the File menu.
    - the print dialog box will appear
3. Press the OK button in the print dialog box.
    - the printer process will begin

All Schedules

1. Open the first page of a project schedule.
2. Select **Print Win/Lib** from the File menu.
   - the print dialog box will appear
3. Press the **OK** button in the print dialog box.
   - the printer process will begin

File
Edit
Go
Hypertools
Flowcharts
Icons
Accessories

CHAPTER 5
Flowchart Processor

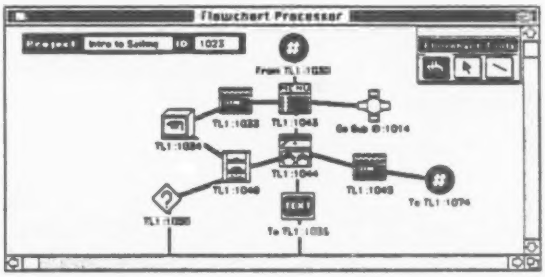
After reading this chapter you will be able to:

—understand the purpose and capabilities of the Flowchart Processor
—create original flowcharts
—edit flowchart pages
—understand the possible relationship between the Flowchart Processor and the Design Templates
—create template pages from the Flowchart Processor

Introduction to the Flowchart Processor

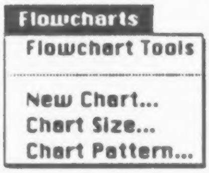
The flowchart processor provides the tools necessary for visualizing the flow of the program. This IDMS tool is part of the ongoing process of designing an interactive program. You can add, delete, or modify items on the flowchart processor as you construct your program.

Icon palettes allow you to select icons to represent each section of the program. The section icons can then be arranged and connected to depict the program's branching. There are two ways to create branching. One option is to click on an icon, which allows you to create and link a representative interactive design template. The other option is to produce an icon from an existing template page and then paste it on a flowchart.



Opening the Flowchart Processor

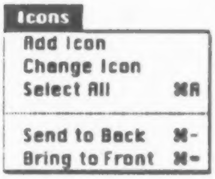
1. Select **Flowchart Processor** (F) from the Hypertools menu.
   - the Flowcharts menu and the Icons menu will be added at the end of the Hypertools menu
   - the flowchart window will then open


Flowcharts
Flowchart Tools
New Chart...
Chart Size...
Chart Pattern...

The Flowcharts Menu

The Flowcharts menu provides options that allow you to create a variety of flowcharts.

Flowchart Tools	displays and hides the Flowchart Tools Palette
New Chart...	creates a new flowchart page in the flowchart window
Chart Size...	allows you to change the size of the currently open flowchart page
Chart Patt...	allows you to choose the background pattern of the currently open flowchart page.

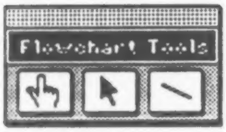

Icons
Add Icon
Change Icon
Select All ⌘A
Send to Back ⌘-
Bring to Front ⌘=

The Icons Menu

The Icons menu allows you to select icons and move them around on a flowchart page.

Add Icon	creates a new icon
Change Icon	changes the existing icon symbol
Select All (⌘A)	selects all the icons on the currently open flowchart page. The Icon Mover Tool can then be used to cut, copy, or move all of the icons simultaneously as a respective group.
Send to Back	sends the selected connecting line(s) or icon(s) behind the unselected objects

Bring to Front	brings the selected connecting line(s) or icon(s) in front of the unselected objects


Flowchart Tools

The Flowchart Tools Palette

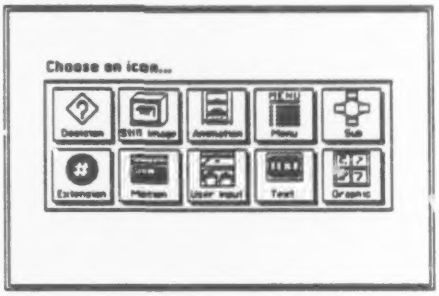
The Flowchart Tools Palette provides three tools that can be used to move between template pages, move icons, or connect icons.

The **Browse Tool** (the hand) is used to go to another template page.

The **Icon Mover Tool** (the arrow) allows an icon to be moved.

The **Icon Connecting Tool** (the line) draws connecting lines between icons.

Adding a New Icon


Choose an icon...
Decision
Still Image
Animation
Menu
Sub
Extension
Motion
User Input
Text
Graphic

1. Open the chart page to which you want to add the icon.

2. Select **Add Icon...** under the Icons menu
   - a dialog box will appear displaying the available icon symbols. The names under the icons serve only as suggestions for their use and will not appear on the flowchart.

3. Select the icon symbol that corresponds to the sequence you are depicting by clicking on it.
   - the dialog box will close

4. Name the icon. Names of 12 or less characters are recommended.

5. Press the **OK** button.

    - the new icon will then automatically appear and will follow the cursor wherever you move it on the flowchart page. Do not drag the icon with the mouse button down. Simply move the mouse and the icon will follow it.

6. Position the icon in the desired location on the flowchart page.

7. Click the mouse to paste the icon.

Linking an Icon to a Template Page

Link an icon to a template page to create a branching network. You can identify icons that have an existing link to a template page because they have the template Page ID Number under them.

1. Choose the **Browse Tool** (the hand) from the Flowchart Tools palette.

2. Click once on the icon that you want to create a template page for.

    - a dialog box will appear

3. Choose the template library where you want to place the template page. Once a template library has been selected, a new template page will be created and the icon's name will be placed in the page name field at the top of the newly created template page. The icon is also assigned the Template Page ID Number in place of the icon name.

Moving Existing Icons

The icon mover (arrow) tool allows you to move icons around the flowchart once they have been pasted. The area in which you can normally move icons is constained to the currently visible window area (the window will not automatically scroll). In order to move an icon beyond the windows borders you must use the special double-click command described below

Moving Icon(s) Within the Borders of the Top Window

1. Choose the **Icon Mover Tool** from the Flowchart Tools palette.
2. Click on and drag the icon to its new position. You can do this with multiple icons simultaneously by holding the shift key down while selecting the icons one-by-one.
    - the icons will appear highlighted and can be moved

Moving an Icon Beyond the Borders of the Top Window

1. Choose the **Icon Mover Tool** from the Flowchart Tools palette.
2. Double click on the icon you wish to move.
    - the cursor will change into the crossed-arrows
    - the icon will then follow the cursor wherever you move it on the flowchart page. Do not drag the icon with the mouse button down. Simply move the mouse and the icon will follow it.
3. Click on the icon to paste it in a new position.

Autoscrolling with the Browse Tool

To automatically scroll the window when moving the **Browse Tool** beyond the borders of the top window, hold the mouse button down.

Template/Flowchart Links

To Travel to a Linked Template Page

1. Choose the **Browse Tool** from the Flowchart Tools palette.
2. Click on the icon which represents the template page you wish to view.
   - the representative template page will then open on top of the flowchart window

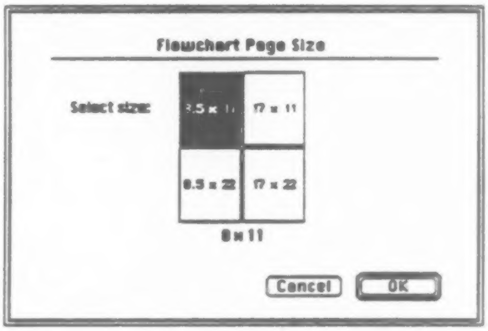
To Return to the Flowchart Page

Click on the **Charted** button on the top of the template page.

Creating a New Chart Page

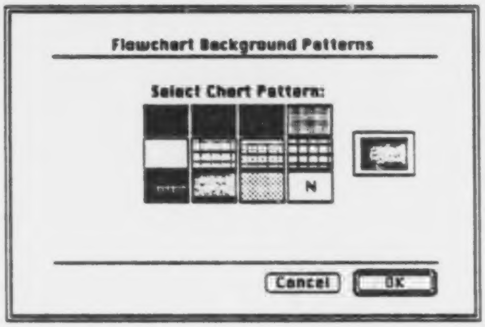
1. Select **New Chart** from the Flowcharts menu.
   - the page size dialog box will appear
2. Select the desired page size.
   - the chart pattern dialog box will then appear
3. Select the desired chart pattern.
   - a new chart page will be created

Changing the Chart Size


Flowchart Page Size
Select size:
8.5 x 11
17 x 11
8.5 x 22
17 x 22
8 x 11
Cancel
OK

1. Select **Chart Size** from the Flowcharts menu.
   - a dialog box will appear
2. Select the desired page size.
3. Press the **OK** button.
   - the chart size will change to the selected size

Changing Chart Background Pattern



1. Select **Chart Patt...** from the Flowcharts menu.
   - a dialog box will appear displaying all the available patterns
2. Click on the desired pattern.
3. Press the **OK** button.
   - the background pattern will change to the selected pattern

Navigating Among Chart Pages

Use the items under the Go menu to move between the different flowchart pages. The right and left arrow keys may also be used to move to the next or previous page.

The Icon Connecting Tool

The Icon Connecting Tool is very similar to the MacDraw line tool. This tool can be used to move connected icons, and it can be used to move connecting lines to new locations.

1. Choose the **Icon Connecting Tool** from the Flowchart Tools palette.
2. Click on the starting icon, hold down the mouse, and drag the cursor to the icon you wish to connect it to.
3. Release the mouse.

Send to Back & Bring to Front Commands

1. Choose the **Icon Mover Tool** from the Flowchart Tools palette.
2. Select the icon and/or connecting line that you wish to move by clicking on it. To select multiple objects, hold the shift key down while clicking on each object.
3. Select **Send to Back** from the Icons menu to place the selected objects behind those that aren't selected.
4. Select **Bring to Front** from the Icons menu to place the selected objects in front of those that aren't selected.

File
Edit
Go
Hypertools
Templates
Find/Sort
Accessories

CHAPTER 6
Interactive Design Templates

After reading this chapter you will be able to:

—understand the purpose and capabilities of the Design Templates
—open the design templates
—understand the different sections of the template page
—create new template pages
—enter information on the template pages
—create branches between template pages
—understand the possible relationship between the Design Templates and the Flowchart Processor
—create flowchart icons from the Design Templates
—understand the purpose and use of Template Libraries
—create new Template Libraries

Using the Design Templates

Template Pages

The design templates can assist you in describing the sequences to be included in the program being designed. Each template page you create begins as a blank form representing one component or sequence of the program. As the program's design matures, more specific information can be added to the page relating to such things as video output, computer output, and user interactivity. Page headings and organization may be customized to fit the specific needs of the project.

Once the template page's branching has been defined, you can pilot the program's interactivity by pressing buttons on the pages. IDMS will automatically shuffle the pages according to your response allowing you to analyze the program's flow prior to production. You may also print out the template pages to perform conventional hard-copy editing.

Template Libraries

When working on large programs (50 pages or more), separate template libraries may be created to organize the different modules or branches of the program. Libraries also allow you to limit template file sizes so that they may be backed-up or transferred onto floppy disks.

Template/Flowchart Interaction

If your design process incorporates both the **Design Templates** and the **Flowchart Processor,** IDMS gives you the capability to combine both of the hypertools and work interactively between them. You can build your design from either direction. When working with the **Design Templates,** you can automatically create a representative flowchart icon. The icon can then be pasted onto a flowchart page, establishing a link between it and the template pages. Alternatively, when working with the **Flowchart Processor,** you may choose to build the design from icons, then automatically create representative template page. Once the representative links are established, you can travel between the two with a click of a button.

Opening the Design Templates

1. Select **Design Templates** from the Hypertools menu.

   - a dialog box will open asking you to select the template library you want to open

2. Select the template library you wish to open by double clicking on the libraries name or select the library and click the **OK** button.

   Note: If you are starting a new project or have not created any additional libraries, the only available library will be named "A. Temp. Lib." Select this library.

   - the Templates and Find/Sort menus will appear

   - the selected template library window will open with the first template page visible

Page Name: Jib Control Intro ☐

Page ID #: TL1: 1000 Created: 4/23/90

Last modified: 4/29/90 by: SAS

Template Library: TEMPLATE LIBRARY 1

Project Name:

Flowcharted: ☒ Chart page#: ID 3509 of

Time: ☒ Estimated ☐ Actual ☐ Non-timed

Dur. : 0:01:15:00 Frames: 2250

Output: ☐ Computer ☐ Video ☒ Both ☐ Other

User Input ☐ No ☒ Yes ☐ Optional Device(s): Mouse

Feedback: ☐ Other >> ☒ Branch

Incoming < Branching > Outgoing

From	User input:	Page ID#	To	User input:	Page ID#
▭	None- Direct		▭	Click Halyard	
▭			▭	Click Mast	
▭			▭	Click Jib	
▭			▭	Click Bow Connection	
▭			▭		
▭			▭		
▭			▭		

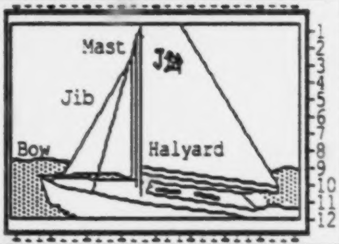
Computer Output: ☐ None ☒ Text ☐ Graphic ☐ Menu ☐ Animation ☐ Other

Type Info: Component Identification

Description: 24pt. Helvetica labels on 1)Halyard, 2) Mast, 3) Jib, and 4) Bow Connection

Type colors: Light Blue: R-205, G-205, B-245

Bkgnd color: Video overlay



Video Output: ☐ None ☒ Video Motion ☐ Video Still ☐ Animation ☐ Text ☐ Other

Scene: Wide shot of bow section

Description: Tape F:34 at 12:14:00
J24 "Lead Sled" heading up with jib tight in.

Location: Lower Superior **Talent:**

Audio: Video Ch.#1: Wind/Music Mix #4 Srce. tape: A7 Talent:

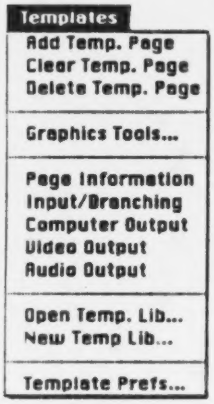
Video Ch.#2: Narration Srce. tape: A2 Talent: Clemons

Script: Computer : Srce. disk:

Used at beginning of sequence:
"Select a section of the jib assembly "

© Interactive Design Managment System 5.1

Synapse Development 89 N. Victoria St.
St.Paul. MN. 55104 (612) 290-0042


Templates
Add Temp. Page
Clear Temp. Page
Delete Temp. Page
Graphics Tools...
Page Information
Input/Branching
Computer Output
Video Output
Audio Output
Open Temp. Lib...
New Temp Lib...
Template Prefs...

The Templates Menu

The Templates menu provides options for manipulating, viewing and organizing the template pages.

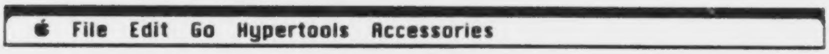
Add Temp. Page	creates a new page
Clear Temp. Page	clears all fields on the template page
Delete Temp. Page	deletes page currently visible

Graphics...	opens the paint tool palette

Page Information	constrains the template window to the page information section of the template page
Input/Branching	constrains the template window to the input/branching section of the template page
Computer Output	constrains the template window to the computer output section of the template page
Video Output	constrains the template window to the video output section of the template page
Audio Output	constrains the template window to the audio output section of the template page

Open Temp. Lib...	allows you to open another template library
New Temp. Lib...	allows you to create another template library

Template Prefs...	the master dialog box where you set template defaults


File
Edit
Go
Hypertools
Accessories

CHAPTER 7
The Accessories

After reading this chapter you will be able to:

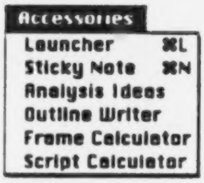
—understand the purpose and capabilities of each of the Accessories
—open each of the Accessories
—use each of the Accessories

Using the Accessories

Similar to the Macintosh's Desk Accessories, IDMS provides the pull-down Accessories menu at all times to aid in the design process. These accessories perform specific functions often required in the design process. You can use the accessories to calculate duration times, add text or audio scripts, make notes to yourself within any document, or a variety of other functions.

Opening the Accessories

1. Select the desired accessory option from the Accessories menu.

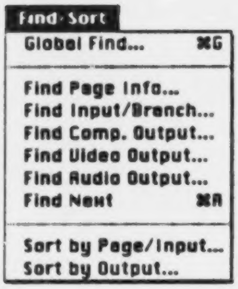

Accessories
Launcher ⌘L
Sticky Note ⌘N
Analysis Ideas
Outline Writer
Frame Calculator
Script Calculator

The Accessories Menu

The Accessories menu accesses the following:

Launcher (⌘L)	opens the Program Launcher palette
Sticky Notes (⌘N)	creates a new Post It™ type Sticky Note
Analysis Ideas	opens the Analysis Ideas Notebook
Outline Writer	opens the Outline Writer word processor

The Find/Sort Menu



The Find/Sort menu provides a listing of specialized searches "finds" and sorts to assist you in locating specific types of data entered ofn the template pages.

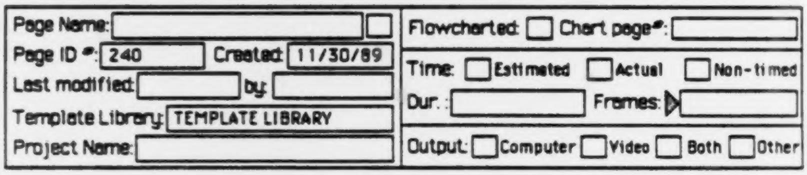
Global Find (⌘ G)	finds a specified word or string stored in any field on the page
Find Page Info...	finds a specified word or string in the information section of the page
Find Input/Branch...	finds a specified word or string in the input/branch section of the page
Find Comp. Output...	finds a specified word or string in the computer output section of the page
Find Video Output...	finds a specified word or string in the video output section of the page
Find Audio Output...	finds a specified word or string in the audio output section of the page
Find Next (⌘ A)	finds the next occurance of a previously specified word or string
Sort by Page/Input...	sorts the template pages according to specified page or input information
Sourt by Output...	sorts the template pages according to specified output information

Sections of the Template Page

Template pages provide a form for recording and organizing detailed descriptions of each program sequence. Each template page is divided into five major sections: page infromation; input/branching information; computer output; video output; and audio output. The various data fields within each of these sections can be used to record specific types of sequence information, as well as to perform various branching funtions. The following is a summary of each of the sections with some suggested uses.

Page Information Section

This section contains fields for recording general information describing the sequence the page represents. The * indicates fields which are automatically filled by IDMS.



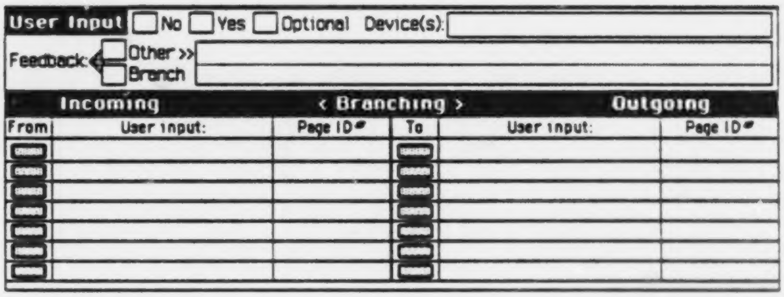
Page Name: ☐
Page ID #: 240 Created: 11/30/89
Last modified: by:
Template Library: TEMPLATE LIBRARY
Project Name:

Flowcharted: ☐ Chart page#:
Time: ☐ Estimated ☐ Actual ☐ Non-timed
Dur. : Frames:
Output: ☐ Computer ☐ Video ☐ Both ☐ Other

Page Name (field)	for recording a page name, specific to that page
Manual Select (box)	a checkbox that can be used to draw special attention to the page (corner folding). Manually selected pages can be found and sorted using the find and sort functions.
Page ID (field)*	contains a uniques ID number which is automatically assigned when page is created
Created (field)*	contains date that page was originally created
Last Modified (field)	for recording the date the page was last modified
By (field)	for recording the name of the user who last made changes
Template Library (field)*	contains the name of the library in which the page resides
Project Name (field)*	computer automatically assigns name from **Project Prefs...** in that box
Charted (button)	a checkbox button which creates a representative flowchart icon when clicked. Once an icon has been created, the box is checked and clicking takes you to the representative icon.

Chart Page (field)*	Once a representative icon has been created, this field contains the name of the flowchart page on which the icon resides.
Estimated Time (box)	for indicating a page for which the actual duration has not been calculated
Actual Time (box)	for indicating a page for which the actual duration has been calculated
Non-timed (box)	for indicating a page that does not have a set duration
Duration (field)	for recording estimated or actual duration times
Frame -> (button)	converts time code values entered in the Duration Field into video frames in the Frames Field when clicked. **Note: Durations most be in this format (2:34:13:20)**
Frames (field)	for recording estimated or actual numbers of video frames. May be calculated automatically by using the Frame Button described above.
Computer Output (box)	for indicating a page containing only computer output
Video Output (box)	for indicating a page containing only video output
Both Output (box)	for indicating a page containing both computer and video output
Other Output (box)	for indicating a page containing an alternative form of output

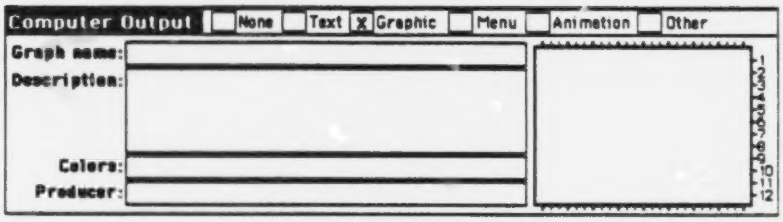
User Input/Branching Section


User Input
☐ No
☐ Yes
☐ Optional
Device(s):
Feedback:
☐ Other >>
☐ Branch
Incoming
< Branching >
Outgoing
From
User input:
Page ID#
To
User input:
Page ID#

No Input (box)	for indicating a page which doesn't require user input
Yes Input (box)	for indicating a page which requires user input
Optional Input (box)	for indicating a page where user input is optional
Device (field)	for describing the type of input device being used
Other (box)	for indicating pages where the feedback is not a branch (eg., computer generated audio or graphics)
Other (field)	for describing non-branching types of feedback (eg., computer generated audio or graphics)
Branch (box)	for indicating pages where the feedback is a branch
From (buttons)	allow you to view any of the previous pages which branch to the current page
From Input (fields)*	describe the input required for each incoming branch to the current page
From Page ID (fields)*	contains the page ID numbers of each of the pages which branch to the current page

To (buttons)	allow you to assign and view the pages the current page branches to
To Input (fields)	for recording the user input required to perform the branch being specified.
	Note: This information must be recorded before using the To button to assign a branch.
To Page ID (fields)*	contains the page ID number of each page the current page branches to

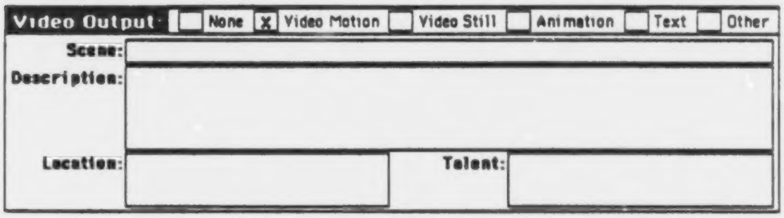
Computer Output Section



Computer Output ☐ None ☐ Text ☒ Graphic ☐ Menu ☐ Animation ☐ Other

Comp. Output (boxes)	for indicating the type of computer output that will be used during this sequence. The selected box dictates what field headings will be used in this page section. These headings may be customised using **Template Preferences**.
Comp. Output (fields)	for recording information related to the design and production of the computer output. The type of information will vary according to how the user customizes the headings.
Screen Design (box)	for storyboarding or creating rough screen layouts with text and graphics

Video Output Section

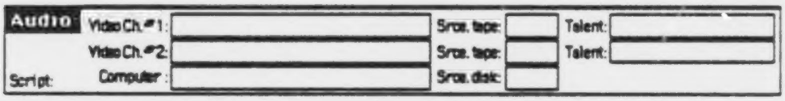


Video Output ☐ None ☒ Video Motion ☐ Video Still ☐ Animation ☐ Text ☐ Other

Scene:
Description:
Location:
Talent:

Video Output (boxes)	for indicating the type of video output that will be used during this sequence. The selected box dictates what field headings will be used in this page section. These headings may be customised using **Template Preferences**.
Video Output (fields)	for recording information related to the design and production of the video output. The type of information will vary according to how the user customizes the headings.

Audio Output Section



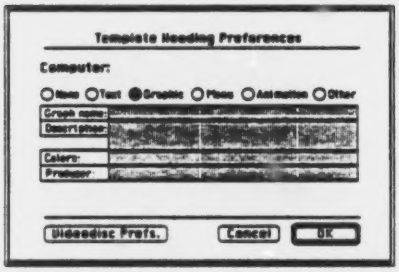
Audio
Video Ch.#1:
Srce. tape:
Talent:
Video Ch.#2:
Srce. tape:
Talent:
Script:
Computer :
Srce. disk:

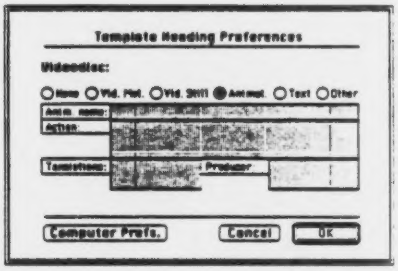
Video Ch. 1 & 2 (fields)	for describing video audio output of each audio channel "track" to be used during this sequence
Source tape Fields 1 & 2	for recording the numbers of the audio or videotapes containing the source material
Talent (fields 1 & 2)	for recording the name of the voice talent used
Computer (field)	for describing video audio output to be used during this sequence
Source disk (field)	for recording the numbers of the disks containing the source material

Script (field)	for recording the narrative script and or decriptions of the soundtrack used during this sequence

Using the Template Pages

There is no "set in stone" method for using the design templates. They are designed to provide you with as many options as possible for recording and viewing design information. There are, however, some important steps and features that require further description.


Template Heading Preferences
Computer:
○ None ○ Text ● Graphic ○ Photo ○ Animation ○ Other
Graph name:
Description:
Colors:
Producer:
Videodisc Prefs.
Cancel
OK


Template Heading Preferences
Videodisc:
○ None ○ Vid. Mot. ○ Vid. Still ● Animat. ○ Text ○ Other
Anim. name:
Action:
Transitions:
Producer
Computer Prefs.
Cancel
OK

Setting Template Preferences

Template Preferences allow you to customize the field headings that are used in the computer and video output sections of the template page to best fit your project's needs. It is best to customize the heading at the onset of a new project. If preference headings are changed later in the project, the new headings will only appear on pages created after the change.

1. Select **Template Prefs.** from the Templates menu.
   - a dialog box will open displaying the computer output headings
2. Click each of the output type boxes to see the default headings for each.
3. Edit or change any or all of the headings to fit your project's needs.
4. Click the **Video Output** button at the buttom left of the dialog box to view the video output section headings.
   - the video output dialog box will open

5. Repeat steps 2 and 3 with the video output headings.

6. Press the OK button to save your heading changes.

Note: If you must change a heading preference after the project's onset, you can get existing pages to reflect the change by reselecting the output type in both the computer and video output sections of each existing page.

Adding a Template Page

1. Open the template library to which you want to add the page.

2. Select **Add Temp. Page** from the Templates menu.

   - a new template page will be added

Deleting a Template Page

1. Open the template page you want to delete. Be sure it is the current page of the current window (on top of everything).

2. Select **Delete Temp. Page** from the Templates menu.

   - the page will be deleted

Note: When deleting or clearing a template page, representative flowchart icons (if exisiting) will not be deleted or cleared. It is a good practice to delete them manually prior to deleting their representive template page.

Clearing a Template Page

1. Open the template page you want to clear. Be sure it is the current page of the current window (on top of everything).

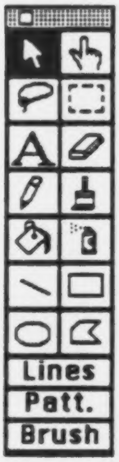
2. Select **Clear Temp. Page** from the Templates menu.

   - all of the pages boxes and fields will be cleared (erased) except for the Page ID#, Project Name, and Template Library Name. The Date Created will be replaced with the current date.

Using the Graphics Palette

The graphic tool palette included in the design templates is designed to be used in the Screen Design Box in the computer output section of the template page. However, they may be used anywhere on the template page. Graphics must be used with extreme caution since they cover the field and button layer ot the page—making it difficult or impossible to enter text in fields covered by graphics.

The graphic tools operate very similary to the MacPaint tools, so only a limited explanation is included here. Graphics created in other programs can be imported and pasted onto the template pages using the Macintosh's clipboard or scrapbook functions.



1. Select **Graphic Tools** from the Templates menu.
   - the Graphic Tools palette will appear
2. Click the graphics tool you wish to use.
   - the selected tool will be highlighted
3. Click and drag the graphic tool to define a "bit box" (drawing surface)
4. Draw and paint your graphic in the bit box.
5. Use the Pointer (arrow) tool to select and move previously drawn bit boxes.
6. Use the Edit menu items to cut, copy and paste selected bit boxes between pages.
7. Close the Graphic Tools palette to return to the text entry and **Browse Tool.**

Constraining the template window

Template pages containing large amounts of information can sometimes become visually overwhelming, especially when you are working on one specific section of the page. Although you can constrain the visible porion of the page by manually resizing the window, the section commands under the Templates menu can make this process quick and easy.

1. Open the template page you wish to work on.

2. Select the section of the page you want to view from the Templates menu.

   - the window will automatically scroll and resize to display only that page section. Go menu commands can be used to browse through the selected page section of all other pages in the currently open library.

Template Branching

By defining branches between template pages, you can later view each branch with the touch of a button. This branching capablility allows you to pilot-test the flow of the entire program before beginning production.

Creating a Template Branch

1. Open the template page you want to branch "from"
2. Locate the topmost line of the Outgoing branching information section that does not already have a branch assigned to it (the page ID field is empty).
3. Type the required user input into the user input field and press return (eg., Keyboard: "upwind").
4. Click on the **To** button which is associated with the input

   - the Template Linker palette will appear
   - the current pages ID and path are automatically placed in the linking palette's original field
5. Open or go to the template page you want the branch to go "to."
6. Press the **This Page** button on the Template Linker palette.

   - the user input, ID and path of the origin page are automatically placed in the fields in the first open line of the current "to" page
   - the origin "from" page reopens
   - the page ID and path of the "to" page is placed in the page ID field of the line from which the branch was initiated

Traveling "to" a Branched Page

Once assigned, the **To** buttons under the outgoing branches section allow you to travel to each page the current page branches "to".

1. Press the **To** button associated with the line (fields) containing the appropriate user input and page ID.
    - the previously linked page will open

Traveling "from" a Branched Page

The **From** buttons under the incoming branches section allow you to travel "backwards" to each page which branch "to" the current page.

1. Press the **From** button associated with the line (fields) containing the appropriate user input and page ID.
    - the previously linked page will open

Deleting a Template Branch

In order to ensure that the proper "to" and "from" branches are deleted, this procedure requires two separate tasks. The first deletes the "to" link and the second deletes the "from" link.

1. Open the template page from which the unwanted branch originates.
2. Locate the branch in the outgoing branches section that you want to delete.
3. Write down the page ID number contained in the branch.
4. Write down the page number of the current page.
5. Hold the option key down on the keyboard and click the **To** button associated with the branch you want to delete.
    - the "to" link will be deleted
    - the associated branching fields will be erased
6. Open the template page the branch went "to" by refering to the page ID# you wrote down in step 3.

7. Locate the line in the incoming branches section which contains the branch "from" the previous page by refering to the page ID# you wrote down in step 4.
8. Hold the option key down on the keyboard and click the **From** button associated with the branch you previously deleted.
   - the "from" link will be deleted
   - the associated fields will be erased

Interaction with the Flowchart Processor

The design templates and the flowchart processor have a unique, interactive relationship. As discussed in Chapter 4, you can create template pages from within the Flowchart Processor to represent individual icons; or, you can create flowchart icons from within the design templates to represent individual template pages. The following section discusses the latter.

Creating representative Flowchart Icons

1. Open the template page for which you want to create a representative icon.
2. Click the Flowcharted box in the page information section of the template page.
   - a dialog box will open asking you to select the symbol for the icon
3. Click the icon symbol you want to use.
   - the Flowchart Processor will open
   - the Flowchart Linker palette will open
4. Open the flowchart page on which you want to place the new icon using the commands from the Go menu.
5. Click the **This Page** button on the linker palete to create the icon.
   - the linker palette will disappear
   - the flowchart tools palette will appear
   - the new icon will appear under the cursor

- the icon will be labeled with the representative template page's ID number
- the cursor will change to the mover cross-arrows
- an X is placed in the Flowcharted box in the page information section of the representative template page to indicate that it has been added to the flowchart
- the flowchart page's ID and path are placed in the Chart Page# field in the page information section of the representative template page

6. Move the cursor (without clicking the mouse) to the position at which you want to paste the icon.
   - the new icon will follow the cursor until the mouse button is clicked
7. Click the mouse button.
   - the icon will be pasted

Note: Refer to Chapter 4 for information on moving, editing, and connecting flowchart icons.

Viewing a representative Flowchart Icon

1. Open the appropriate template page
2. Click the Flowcharted box in the page information section of the template page.
   - the appropriate flowchart page will open
   - the representative icon will flash twice to identify its location

Returning to the Representative Template Page

1. Choose the **Browse Tool** from the Flowchart Tools palette.
2. Click on the icon which represents the template page you wish to view.
   - the representative template page will then open on top of the flowchart window

Deleting a Representative Template Icon

1. Open the appropriate template page.
2. Click the Flowcharted box in the page information section
   - a flowchart page will open and the representative icon will flash twice to identify its location
   - the Flowchart Tools palette will appear
3. Select the Icon Mover Tool (arrow) from the Flowchart Tools palette.
4. Single click the representative flowchart icon.
   - the icon will be selected
5. Select Cut from the Edit menu.
   - the icon will be deleted
6. Reopen the representative template page.
7. Hold the option key down while clicking the Flowcharted box.
   - the icon link will be deleted
   - the X in the Flowcharted box will disappear
   - the chart page# field will be cleared

Using Template Libraries

Template libraries can be used to organize different sections of a program's design. When using multiple libraries, two or more libraries may be opened simultaneously for easy cross referencing. You should keep each template library under 780K. This ensures that floppy disks will be able to store the amount of information you want to copy onto them.

Creating a New Template Library

1. Select **New Temp. Lib...** from the Templates menu.
2. Open the most recently created template library (eg., if you have 5 libraries, then open library #5).
3. Name the new library.
   - a new library will be created and opened

Opening a Template Library

1. Select **Open Temp. Lib...** from the Templates menu.
    - a dialog box appears
    - two new menu items will appear at the top of the screen after tools: Templates and Find/Sort. Each one creates a new menu.
2. Click on the library you want.
3. A window opens displaying the first page of that template library.

Navigating through theTemplate Pages

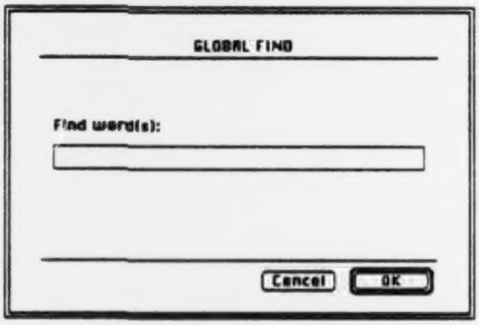
Once you've created a series of template pages, you will need to be able to easily move between them. The keyboard's arrowkeys and the commands under the Go menu can aid you in moving from one page to the next and jumping to the first and last pages. However, as your project begins to grow and more and more pages are added to the template libraries, use of the commands under the Find/Sort menu will become a necessity.

Find commands allow you to find a specific word or string of words in a specific field on a template page. This can be used to find pages by name, ID number, input, output or a variety of other categories. The **Global Find** command is less specific, locating template pages containing the specified word or string of words anywhere on the page. Once one occurrence has been found, the **Find Next** command can be used to find the next occurrence.

Sort commands give you the ability to sort and group the template library's pages in a variety of different ways. IDMS automatically orders "sorts" pages by their ID number when a new project is started or a new library is created. The Sort commands allow you to re-sort the pages according to the information contained in a specific field. This command can be used to collect and place up front all pages of a specific type. Some typical uses include collecting non-flowcharted pages, non-branched pages, and sorting by types of media.

Global Finds

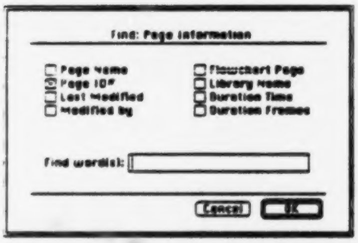
Global finds allow you to find the occurrence of a word or string of words in any field of any template page.

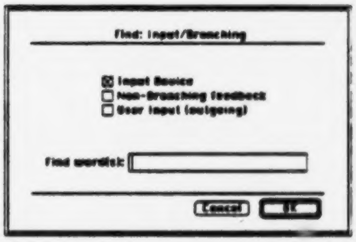


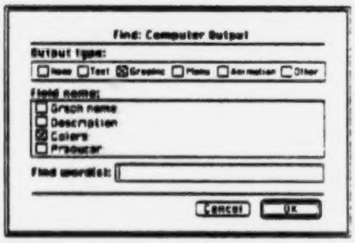
1. Select **Global Find** from the Find/Sort menu.
   - a dialog box will open asking you to specify the word(s) you are trying to find
2. Enter the word or string of words you want to find.
3. Click the **OK** button.
   - the Find Progress dialog box will appear
   - the pages will be searched until the word(s) is found
   - the first page containing the specified word(s) will be displayed with an audible beep
   - no beep will be heard if no occurrences are found
4. Select **Find Next** from the Find/Sort menu to find the next occurrence of the word(s).

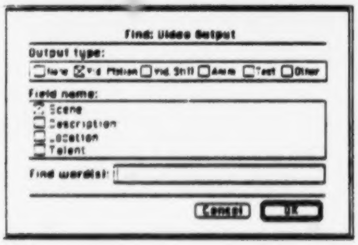
Field Specific Finds

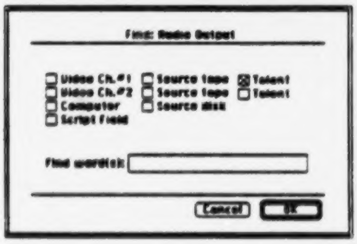
Field specific finds allow you to find the occurrence of a word or string of words in a specific field. There is a separate field specific find command for searching each section of the template page. For example, if you wanted to find a video motion sequence called "Folding the sails", you would select **Find Video Output** from the Find/Sort menu. The following is a summary of the standard procedure for performing a field specific find.

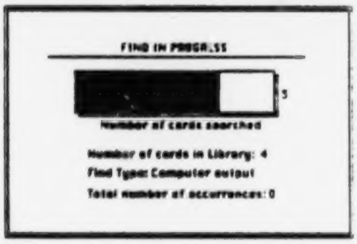

Find: Page Information
☐ Page Name
☒ Page ID#
☐ Last Modified
☐ Modified by
☐ Flowchart Page
☐ Library Name
☐ Duration Time
☐ Duration Frames
Find word(s):
Cancel
OK


Find: Input/Branching
☒ Input Device
☐ Non-Branching feedback
☐ User Input (outgoing)
Find word(s):
Cancel
OK


Find: Computer Output
Output type:
☐ None
☐ Text
☒ Graphic
☐ Menu
☐ Animation
☐ Other
Field name:
☐ Graph name
☐ Description
☒ Colors
☐ Producer
Find word(s):
Cancel
OK


Find: Video Output
Output type:
☐ None
☒ Vid. Motion
☐ Vid. Still
☐ Anim
☐ Text
☐ Other
Field name:
☒ Scene
☐ Description
☐ Location
☐ Talent
Find word(s):
Cancel
OK


Find: Audio Output
☐ Video Ch.#1
☐ Video Ch.#2
☐ Computer
☐ Script Field
☐ Source tape
☐ Source tape
☐ Source disk
☒ Talent
☐ Talent
Find word(s):
Cancel
OK


FIND IN PROGRESS
5
Number of cards searched

Number of cards in Library: 4

Find Type: Computer output

Total number of occurrences: 0

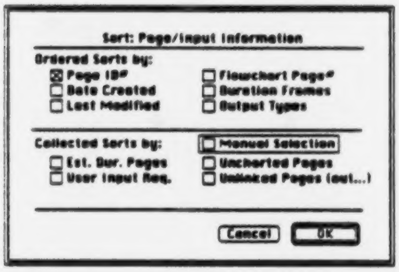
1. Select the find command associated with the template page section you want to search from the Find/Sort menu.
   - a dialog box will open asking you to specify the field you want to search and the word(s) you are trying to find
2. Select the media type using the check boxes (if available).
3. Select the the field to search using the check boxes.
4. Enter the word or string of words you want to find.
5. Click the **OK** button.
   - the Find Progress dialog box will appear
   - the pages will be searched until the word(s) is found
   - the first page containing the specified word(s) will be displayed with a audible beep
   - no beep will be heard if no occurrences are found

6. Select **Find Next** from the Find/Sort menu to find the next occurrence of the word(s).

Ordered and Collected Sorts

IDMS allows you to perform two different types of sorts depending upon the type of information you are working with. Ordered sorts will sort the pages alphabetically or numerically according to the type of data field you sort by. For example, you could sort the pages by the dates on which they were created or the length of their durations. Collected sorts, on the otherhand, allow you to find and collect all the pages containing (or not containing) a certain type of information and place all of them together in the front of the template library. This can be used to do such things as collect all the pages where user input is required, or where video animation is used.

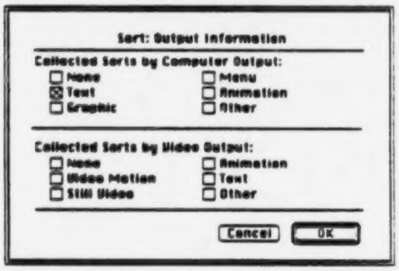
The Manual Selection sorting function contained in the Page/Input sort dialog box allows you to mark individual pages within a template library and then bring (collect) all the marked pages to the front of the library. Pages are marked by clicking the Manual Selection box in the top center of the page information section of the template page. This can be used to collect pages that need work or special attention.



Sort: Page/Input Information

Ordered Sorts by:
☒ Page ID#
☐ Date Created
☐ Last Modified
☐ Flowchart Page#
☐ Duration Frames
☐ Output Types

Collected Sorts by:
☐ Manual Selection
☐ Est. Dur. Pages
☐ User Input Req.
☐ Uncharted Pages
☐ Unlinked Pages (out...)



Sort: Output Information

Collected Sorts by Computer Output:
☐ None
☒ Text
☐ Graphic
☐ Menu
☐ Animation
☐ Other

Collected Sorts by Video Output:
☐ None
☐ Video Motion
☐ Still Video
☐ Animation
☐ Text
☐ Other

The **Sort by Page/Input...** item from the Find/Sort menu can perform both ordered and collected sorts while the **Sort by Output...** item only performs for collected sorts. The following is a summary of the standard procedure for performing a sort.

1. Select sort command associated with the template page section you want to search from the Find/Sort menu.
    - a dialog box will open asking you to specify the type of sort
2. Select the box associated with the type of information you want to sort by.

3. Click the **OK** button.
    - the Sort Progress dialog box will appear
    - the pages will be sorted by the specified field information
    - the pages will be reordered (after an ordered sort)

    or

    - the collected pages will be placed at the front of the library (after a collected sort)

Printing Template Pages

Printing a single Template Page

1. Open the template page you want to print.
2. Select **Print Page** from the File menu.
    - the printing dialog box will open
3. Set up the print parameters.
4. Click the **OK** button.
    - the template page will be printed

Printing a Template Library

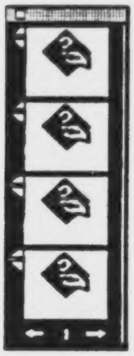
1. Open the template library you want to print.
2. Select **Print Window** from the File menu.
    - the printing dialog box will open
3. Set up the print parameters.
4. Click the **OK** button.
    - all the pages in the template library will be printed

Note: This page has been intentionally left blank to maintain the indexing of the original software manual within this thesis appendix.

Frame Calculators	opens the Frame Calculator palette
Script Calculator	opens the script calculator palette

Using the Program Launcher

The Program Launcher allows you to launch external applications without having to return to the finder. You can customize the launcher to launch up to twelve different applications. The Program Launcher can be used as an efficient method of opening an external application, copying some text or graphics, and then bringing them into IDMS for pasting.



Adding Applications to the Launcher

1. Open the Launcher palette by selecting Launcher from the Accessories menu.
   - the Program Launcher palette will open
2. Click on a launcher button that has not yet been assigned (one containing an application symbol with a question mark).
   - a dialog box will open asking you to select the application
3. Double click on the application you want to add to the launcher.
   - a dialog box will open asking you to select an icon
4. Click on the icon which best represents the application title.
   - the selected icon and application name will appear on the Program Launcher palette
   - the application is ready to be launched

Launching a Launcher Application

1. Open the Launcher palette by selecting Launcher from the Accessories menu.

2. Click on the launcher button representing the application you want to launch

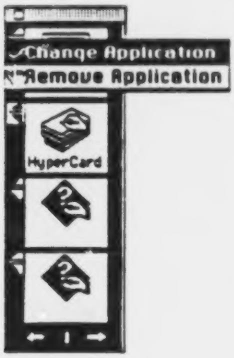
   - IDMS will close (saving changes)

   - the application will open

Returning to IDMS from a Launched Application

Quit the application using the applications Quit command

- IDMS will reopen automatically



Removing a Launcher Application

1. Open the Launcher palette by selecting Launcher from the Accessories menu.

2. Click and hold down on the small black triangle to the upper-left of the application button you want to delete.

   - a pop-up menu will appear displaying two options

3. Still holding the button down, slide the mouse towards the option "Remove Application" until it highlights (turns black).

4. Release the mouse button

   - the application button will be cleared and re placed with the unassigned icon

Changing a Launcher Application

1. Open the Launcher palette by selecting Launcher from the Accessories menu.

2. Click and hold down on the small black triangle to the upper-left of the application button you want to delete.

   - a pop-up menu will appear displaying two options

3. Still holding the button down, slide the mouse towards the option "Change Application" until it highlights (turns black).

4. Release the mouse button.

   - the previously assigned application will be removed

   - a dialog box will open asking you to select the new application

5. Double click on the application you want to add to the launcher.

   - a dialog box will open asking you to select an icon

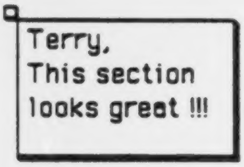
6. Click on the icon which best represents the application title.

   - the selected icon and application name will replace the previous application on the Program Launcher palette

   - the new application is ready to be launched

Using Sticky Notes

Sticky Notes allow you to paste Post It™ type notes to any IDMS document. They can be used for such things as reminders and recording details. If a Sticky Note is left "open" it will be visible on printed documents.

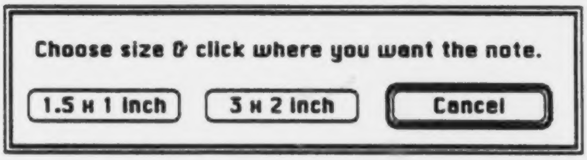


Terry,
This section
looks great !!!

Creating a Sticky Note

1. Select **Sticky Notes** from the Accessories menu.

   - a dialog box appears



Choose size & click where you want the note.

2. Choose one of the four note sizes by clicking on the one you want.
    - the dialog box disappears

3. Click on the document where you want the upper left corner of the **Sticky Note** to appear.

Note: A sticky note is represented by a shadowed box. Click on that and a field appears to enter information on.

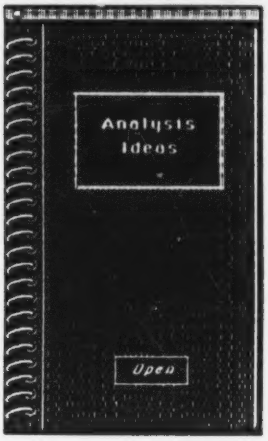
Deleting a Sticky Note

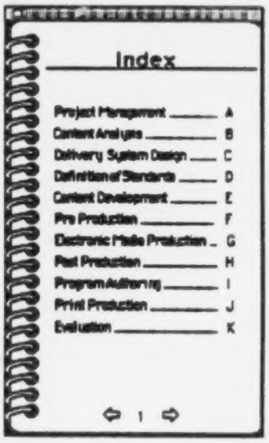
Hold down the option key and click on the shadowed note box.

- the **Sticky Note** disappears

Using Analysis Ideas

Analysis Ideas is a computer-based notebook containing a series of questions and ideas related to interactive videodisc design and project management. It can be used alone as a reference Accessory or together with the Outline Writer as the Analysis Outliner Hypertool (refer to chapter 3).





Index

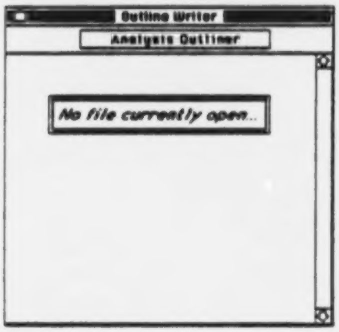
Project Management	A
Content Analysis	B
Delivery System Design	C
Definition of Standards	D
Content Development	E
Pre Production	F
Electronic Media Production	G
Post Production	H
Program Authoring	I
Print Production	J
Evaluation	K

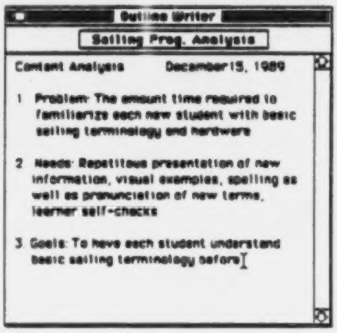
Opening Analysis Ideas

1. Select **Analysis Ideas** from the Accessories menu.
   - the Analysis Ideas notebook will open
2. Click the **Open** button on the notebooks cover.
   - the contents page will appear
3. Click on the chapter's title you wish to view.
   - the specified chapter will open
4. Use the **<-** and **->** buttons at the bottom of the pages to "browse" forward and backward through the chapter's pages.
5. Click the **Content** button on the bottom of the page to return to the table of contents page.

Using the Outline Writer

The Outline Writer is a simplified word processor for writing and saving information. It is capable of exporting and importing text documents to and from other applications. The Outline Writer can be used alone as a notetaking and reference Accessory or together with Analysis Ideas as the Analysis Outliner Hypertool (refer to chapter 3).





Content Analysis December 15, 1989

1 Problem: The amount time required to familiarize each new student with basic sailing terminology and hardware

2 Needs: Repetitious presentation of new information, visual examples, spelling as well as pronunciation of new terms, learner self-checks

3 Goals: To have each student understand basic sailing terminology before

Opening the Outline Writer

Select **Outline Writer** from the Accessories menu.

- the Outliner menu will appear
- the Outline Writer window will open

Creating a New Outliner File

1. Select **New Outline File...** from the Outliner menu.
   - a dialog box will open asking you to name the new file
2. Enter a file name and click the **OK** button in the dialog box.
   - a new Outline Writer page will appear with the file name displayed in the header
3. Type your document in the Outline Writer window.
   - all entries and changes will automatically be saved when the Outline Writer window is closed or another outline file is opened

Opening an Existing Outliner File

1. Select **Open Outline File...** from the Outliner menu.
   - a dialog box will open asking you to select the file you want to open
2. Double click on the file name or select it and click the **Open** button to open the file.
   - the selected file will appear in the Outline Writer window

Deleting an Outliner File

1. Open the outliner file you want to delete.
2. Select **Delete Outline File...** from the Outliner menu.
   - a dialog box will open asking you if you want to delete the file
3. Click the **OK** button to delete the file or the **Cancel** button to abort the delete.

Importing a Text File

1. Select **Import Text File** from the Outliner menu.
    - a dialog box will open asking you to select the file you want to import
2. Double click on the name of file you want to open or select the file's name and click the **Open** button.
    - a copy of the file will open in the outline writer window and the name of the file will be dis played in the heading. The original text file will still exist outside IDMS.

Exporting a Text File

1. Open the outline file you wish to export.
2. Select **Export Text File...** from the Outliner menu.
    - a dialog box will open asking you to name the new text file
3. Rename the file or leave the name the same as the Outliner file.

    Note: Since text files reside outside of the IDMS system, they may have the same name as outliner files.

4. Click the **Save** button.
    - a text file is created in the IDMS folder

Changing the Font

The Font menu allows you to change the font of any selected text in the Outline Writer window. These font changes only effect the text displayed on the screen and will not have any effect on exported text files or files printed from within IDMS.

1. Select the text you want the font change to effect.
2. Select the new font from the Font menu.
    - the font of the selected text will be changed

Printing Outliner Files

In most cases it is recommended that you export, edit and print important outliner files with a standard word processor since it will allow you greater formatting capabilities. However, you can print a hard copy of any or all of the Outliner documents using the **Print Page** and **Print Win/Lib** commands.

To print a single a outliner file:

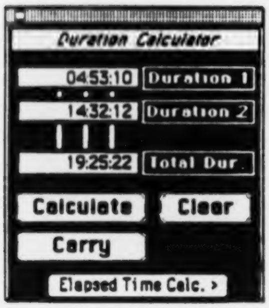
1. Open the Outline Writer file you wish to print.
2. Select **Print Page** from the File menu.
    - the print dialog box will open

To print all outliner files:

1. Open the Outline Writer.
2. Select **Print Win/Lib** from the File menu.
    - the print dialog box will open

Using the Frame Calculators

The Frame Calculators are an efficient way of adding, subtracting, and converting time code values. This can be very useful in calculating disc space from template page information.



04:53:10 Duration 1
14:32:12 Duration 2
19:25:22 Total Dur.

The Duration Calculator

The Duration Calculator allows you to calculate the duration between two timecode values and then convert the duration into video frames.

1. Select **Frame Calculator** from the Accessories menu.
    - the Frame Calculator appears. It is defaulted to the Duration Calculator
2. Click on the **Entry In** button.
    - the cursor goes to this field
3. Enter a value for **Entry In**. Values must be entered in minutes, seconds, and frames per second (eg., 12:00:29). Video has 30 frames per second. Separate time periods with colons or semi-colons.

4. Repeat steps 2 and 3 to enter a value for **Entry Out**.

5. Press the **Calculate** button to calculate the duration times and video frames.

   - The value for the duration time will appear in the Duration field, and the value for the video frames will appear in the Frames field.

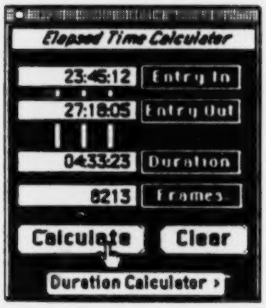
Note: When converting durations to just video frames, you do not need to enter Entry In and Entry Out values.

6. To clear a duration field, click on the button title of the desired field to be cleared (eg., **Entry In**). To clear all fields, press the **Clear** button.

   - the fields are cleared (erased)

7. To change calculators, press on the **Add Durations >** button.

   - the Elapsed Time Calculator appears



23:45:12 Entry In
27:18:05 Entry Out
04:33:23 Duration
8213 Frames

The Elapsed Time Calculator

The Elapsed Time calculator allows you to calculate the sum of two time code durations.

1. Select **Frame Calculator** from the Accessories menu.

   - the Frame Calculator appears. It is defaulted to the Duration Calculator

2. Click on the **Add Durations >** button to change to the Elapsed Time Calculator.

3. Click on the **Duration 1** button.

   - the cursor goes to this field

4. Enter a value for **Duration 1**. Values must be entered in minutes, seconds, and frames per second (eg., 36:59:08). Video runs at 30 frames per second. Separate time periods with colons or semi-colons.

5. Repeat steps 2 and 3 to enter a value for **Duration 2**.

6. Press the **Add** button to add the two duration segments.

   - the value will appear in the **Total Dur. Field**

7. To add another duration segment, press the **Carry** button.

   - the **Carry** function takes the value in **Total Dur.** and puts it in the **Duration 1** field. You can then repeat steps 5 and 6 for a new value. You may use the **Carry** function as many times as needed.

8. To clear a duration field, click on the button title of the desired field to be cleared (eg., **Duration 1**). To clear all fields, press the **Clear** button.

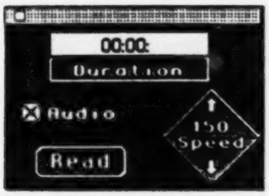
   - the fields will be cleared (erased)

9. To change calculators, press on the **Calc. Frames >** button.

   - the Duration Calculator reappears

Using the Script Calculator

The **Script Calculator** estimates the duration time of audio output. It estimates the duration in actual time. This means that you must wait until the entire text has been read to continue working (eg., If it takes you ten minutes to actually read the text, then the **Script Calculator** will take ten minutes to read the text). This calculator can be used with any task in any document.



Note: You must have Apple's Macintalk driver residing in your system folder in order for the calculator to function. Due to diffenences in new hardware and system software, the Script Calculator may not function properly on some systems.

1. Select **Script Calculator** from the Accessories menu.

   - the **Script Calculator** appears

2. Click on and drag the mouse to highlight the text that you want the **Script Calculator** to read.

- IDMS calculates the minutes and seconds of the audio duration. It does not calculate frames per second.

Adjusting the Narration Speed

1. To increase the narration speed, press the upper arrow on the narration speed indicator. (graphic)
    - the increased speed will appear on the indicator
2. To decrease the narration speed, press the lower arrow on the narration speed indicator.
    - the decreased speed will appear on the indicator

Reading the Script

1. To hear your script, press the **Read** button.
    - the computer reads your audio output
    - the audio output clears itself when you read something else
    - the **Script Calculator** can be used with the Duration Calculator to add segments of output
2. To turn the sound off, press the **Audio** button.

CHAPTER 8
Developing an IDMS work procedure

After reading this chapter you will be able to:

- —understand the benefits of IDMS's non-linear design
- —apply various time-saving tips and techniques when working on an IDMS project
- —understand one of the many design procedures possible with IDMS

What is the best work procedure?

IDMS is a non-linear software program. Because of this, the system itself does not dictate the process by which you work. When you open IDMS you are presented with menus rather than the opening of a specific tool, allowing you to chart the path which best fits your project needs. Even once a project has been started, you can move between tools in any order you choose or open two or more simultaneously.

When starting a new project, the procedural options may seem overwhelming. The IDMS model (refer to the project schedules or appendix I) can be used to help develop your initial procedure. Once you've started you can still change you mind, and jump between all of the different tools in any order you wish. Remember, the purpose of IDMS is to allow you to work interactively, not linearly. This will take some getting used to, but after while you will discover how to make this flexibility work to your advantage.

Assorted tips and techniques

The following is a list of tips, techniques, and suggestions to consider any time you are working with IDMS.

1. Keep windows as small as possible to speed up processing time and reduce visual distraction.
2. Close windows not currently in use to speed up processing time.

3. Open and reduce multiple windows for easy cross referencing between documents (flowcharts & templates, or templates and outline documents.
4. Print out any documents you need to refer to often.
5. Use the Save command after making many changes to speed up processing time.
6. Use the Macintosh's clipboard to bring in graphics and text from other applications.
7. Use the Outline Writer Accessory to write and edit large blocks of text, then cut and paste onto templates etc.
8. Use Sticky Notes to indicate problem areas, and write comments.
9. Review your analysis documents periodically to evaluate your progress.

An outline of one possible work procedure

The following work procedure is provided as an example of one possible way to integrate the components of the IDMS system. This particular approach progresses from general to macro to micro design procedures. Since the operation of each of the system's tools and components is detailed in the index and body of this manual, very little detail is provided here on system operation. Items marked with an " * " indicate additional options or suggestions.

Note: The disadvantage in providing this type of example is that it can not depict the interactive building that is done constantly throughout the design process. This example lists (linearly) only the major design steps and does not fully depict the constant , non-linear, cross-referencing involved in the process.

1. Install IDMS on your computer's hard disk
2. Set the project preferences
3. Use the Analysis Outliner to:
   - begin to focus on your design by answering the questions in Chapters 1 and 2 of the Analysis Ideas notebook
   - work on one chapter at a time
   * save your notes for each chapter as a different outliner file for easy referencing
   * print out a copy for brainstorming on paper

4. Use the Project Schedules to:
   - edit the tasks to fit the needs of your current project
   - enter estimated starting and completion dates for schedule pages 1-5
   - assign staff names to design tasks

5. Continue using the Analysis Outliner to:
   - focus your design by answering the questions in Chapters 3 through 5 of the Analysis Ideas notebook

6. Use the Launcher Accessory to:
   - launch necessary applications outside of IDM

7. Use the Flowchart Processor to:
   - identify major program section and create a different flowchart page(s) for each
   - rough out sections of your design with icons and sequence names
   - * use Sticky Notes to carry more info
   - use the connecting tool to roughly depict flow

8. Once you are comfortable with a section of the flowchart begin to create representative Design Template pages from the Flowchart icons
   - click on a flowchart icon with browse tool to create a Template page
   - * create different template libraries for each major program section. This must be done from within the Design Templates

9. Continue using the Analysis Outliner to:
   - begin planning the rest of the program's production by answering the questions in chapters 6 through 11 of the Analysis Ideas notebook

10. Continue using the Project Schedules to:
    - input and update task related dates
    - * print out updated schedules for progress reports
    - enter estimated starting and completion dates for schedule pages 6-11
    - assign task links between task and related design documentation

11. Continue using the Template pages to:
    - set template heading preferences
    - begin detailed description of required inputs and outputs
    - begin to depict interpage branching
    - * open both the flowchart page and template page windows to easily cross reference the necessary branching
12. Continue using the Flowchart Processor to:
    - complete flowcharting all remaining program section
    - create representative Template pages
    - * print out entire flowchart and post for easy referencing
    - * delete all old Sticky Notes
13. Continue using the Template pages to:
    - complete detailed description of required inputs and outputs
    - complete to depict interpage branching
    - * use the Find/Sort commands to locate

APPENDIX I
Interactive Design Management Model

A. PROJECT MANAGEMENT

1. Define staffing requirements
2. Identify project team
3. Project team meeting
4. Establish standard reporting procedures
5. Establish standard for documenting program updates and changes
6. Establish system fo change authorization
7. Define critical dates and deadlines
8. Develop work status estimating, reporting, and monitoring procedure
9. Develop project work schedule
10. Develop formative quality assurance procedures
11. Develop individual work schedules
12. Develop integrated team work schedules
13. Develop file sharing and organization procedures
14. Establish problem reporting and resolution procedures
15. Estimate resources and materials required
16. Perceived cost/benefit analysis
17. Assess available budget

B. CONTENT ANALYSIS

1. Identify program topic
2. Conduct needs assessment
3. Define rationale for program
4. Define program goals
5. Define program scope
6. Identify primary, secondary audiences
7. Define audience characteristics
8. Define learning environment
9. Define program user prerequisites
10. Define content assumptions
11. Determine internal evaluation criteria
12. Determine external evaluation criteria
13. Develop behavioral objectives
14. Develop measurable objectives
15. Define instructional strategies
16. Define methods of measurement and evaluation
17. Define number of program copies to be produced

C. DELIVERY SYSTEM DESIGN

1. Define hardware configuration
2. Choose authoring system or language
3. Define authoring capabilities
4. Define user interface(s)

D. DEFINITION OF STANDARDS

1. Define program defaults
2. Define intended affect on viewer
3. Define programs personality
4. Define feeling of programs environment
5. Define production values
6. Define video consistency standards
7. Define computer graphic design consistency standards
8. Define transition consistency standards
9. Define screen design consistency standards
10. Define menu usage consistency standards
11. Define icon usage consistency standards
12. Define color coding and usage consistency standards
13. Define narrative consistency standards
14. Define video sound effect consistency standards
15. Define video soundtrack usage consistency standards
16. Define video text consistency standards
17. Define print material consistency standards
18. Define computer generated audio consistency standards
19. Define computer generated text consistency standards
20. Define touch screen areas
21. Define types of user interactivity
22. Define input device usage consistency
23. Define learner tracking strategy

E. CONTENT DEVELOPMENT

1. Identify subject matter experts
2. Select subject matter experts
3. Conduct subject matter analysis
4. Interview subject matter experts
5. Conduct task analysis
6. Generate rough content outline
7. Review existing content materials
8. Write new content components
9. Modulize content components
10. Identify points of formative user evaluation
11. Pair evaluation points with objectives
12. Develop concept flowchart

13. Define branching alternatives
14. Define detailed content outline
15. Identify types of print materials
16. Outline print materials
17. Write first draft of narrative script
18. Define inner module interactivity
19. Select specific mediums for each program segment
20. Create detailed content flowchart
21. Pilot test content flow
22. Devise user introduction to system operation
23. Team evaluation of design
24. Revise design
25. Acceptance of final design
26. Create design document

F. PRE-PRODUCTION

1. Estimate video production budget
2. Identify existing footage & sources
3. Create video shot list
4. Define video treatment
5. Write location descriptions
6. Write set descriptions
7. Indentify props needed
8. Write still frame descriptions
9. Create still frame sketches
10. Write descriptions of graphics
11. Write video effects descriptions
12. Write text frame descriptions
13. Write menu descriptions
14. Write animation descriptions
15. Create animation storyboards
16. Write stage directions
17. Identify studio requirements
18. Evaluate shot locations
19. Collect production bids
20. Secure prod. house and crew
21. Make location & studio arrangements
22. Generate final narrative script
23. Create teleprompter script
24. Create voice over script
25. Design disc audio track usage
26. Estimate completed disc(s) length
27. Identify necessary talent
28. Audition talent

F. PRE-PRODUCTION (cont.)

29. Select talent
30. Identify soundtrack styles
31. Identify sources of soundtrack mat.
32. Team review of script and storyboards
33. Create production schedules
34. Create production reports
35. Rehearsal

G. ELECTRONIC MEDIA PRODUCTION

1. Production crew meeting
2. Shoot studio video sequences
3. Shoot location video sequences
4. Produce still frames
5. Record studio audio
6. Record original music
7. Obtain "needle drop" music
8. Produce computer graphics
9. Produce animation sequences
10. Produce video text/menu frames
11. Review with design team
12. Create post-production document

H. POST PRODUCTION

1. Create window burn dubs of all source reels
2. Log window burn tapes
3. Create a disc map
4. Create rough edit decision list
5. Offline edit rough edit tape
6. Review offline edit tape
7. Revise edit decision list
8. Recalculate disc space usage
9. Online edit
10. Create dubs of pre-master reels
11. Review premaster
12. Revise and review pre-master reel if needed
13. Print final edit decision list
14. Produce check disc
15. Review check disc
16. Revise pre-master if necessary
17. Log check disk frame numbers
18. Replicate final discs

I. PROGRAM AUTHORING

1. Orient software development team
2. Finalize software design
3. Outline program defaults and standards
4. Log frame #s in program flowchart
5. Author course sequences
6. Develop formative software testing
7. Interface software with check disc
8. Pilot test software
9. Log all bugs and inconsistencies
10. Revise software
11. Outline software operation guidelines
12. Replicate software

I. PRINT PRODUCTION

1. Orient graphic designers
2. Orient technical writers
3. Write draft of copy to be used
4. Revise copy to be used
5. Create possible design thumbnails
6. Choose best design(s)
7. Identify photo and design needs
8. Create design roughs
9. Create all necessary graphics
10. Obtain all necesary photos
11. Obtain printing bids
12. Identify typesetter and printer
13. Set type for all copy
13. Scan or paste-up photo/graphics
14. Obtain comprehensive copy
15. Pilot the print materials
16. Revise and review print materials
17. Reproduce print materials

K. EVALUATION

1. Develop program evaluation documents
2. Review program evaluation documents
3. Revise program evaluation documents
4. Create pre-tests
5. Review pre-tests
6. Revise pre-tests
7. Create post-tests
8. Review post-tests
9. Revise post-tests

Note: This page has been intentionally left blank to maintain the indexing of the original software manual within this thesis appendix.

APPENDIX II
Blank IDMS Forms

Contains:

1. Design Template Page
2. Project Schedule Page (with date fields)
3. Project Schedule Page (with staff fields)

Note: This page has been intentionally left blank to maintain the indexing of the original software manual within this thesis appendix.

Page Name: ☐

Page ID #: Created:

Last modified: by:

Template Library:

Project Name:

Flowcharted: ☐ Chart page#:

Time: ☐ Estimated ☐ Actual ☐ Non-timed

Dur. : Frames:

Output: ☐ Computer ☐ Video ☐ Both ☐ Other

User Input ☐ No ☐ Yes ☐ Optional Device(s):

Feedback: ☐ Other >> ☐ Branch

Incoming			< Branching >		Outgoing
From	User input:	Page ID#	To	User input:	Page ID#

Computer Output: ☐ None ☐ Text ☐ Graphic ☐ Menu ☐ Animation ☐ Other

1 2 3 4 5 6 7 8 9 10 11 12

Video Output: ☐ None ☐ Video Motion ☐ Video Still ☐ Animation ☐ Text ☐ Other

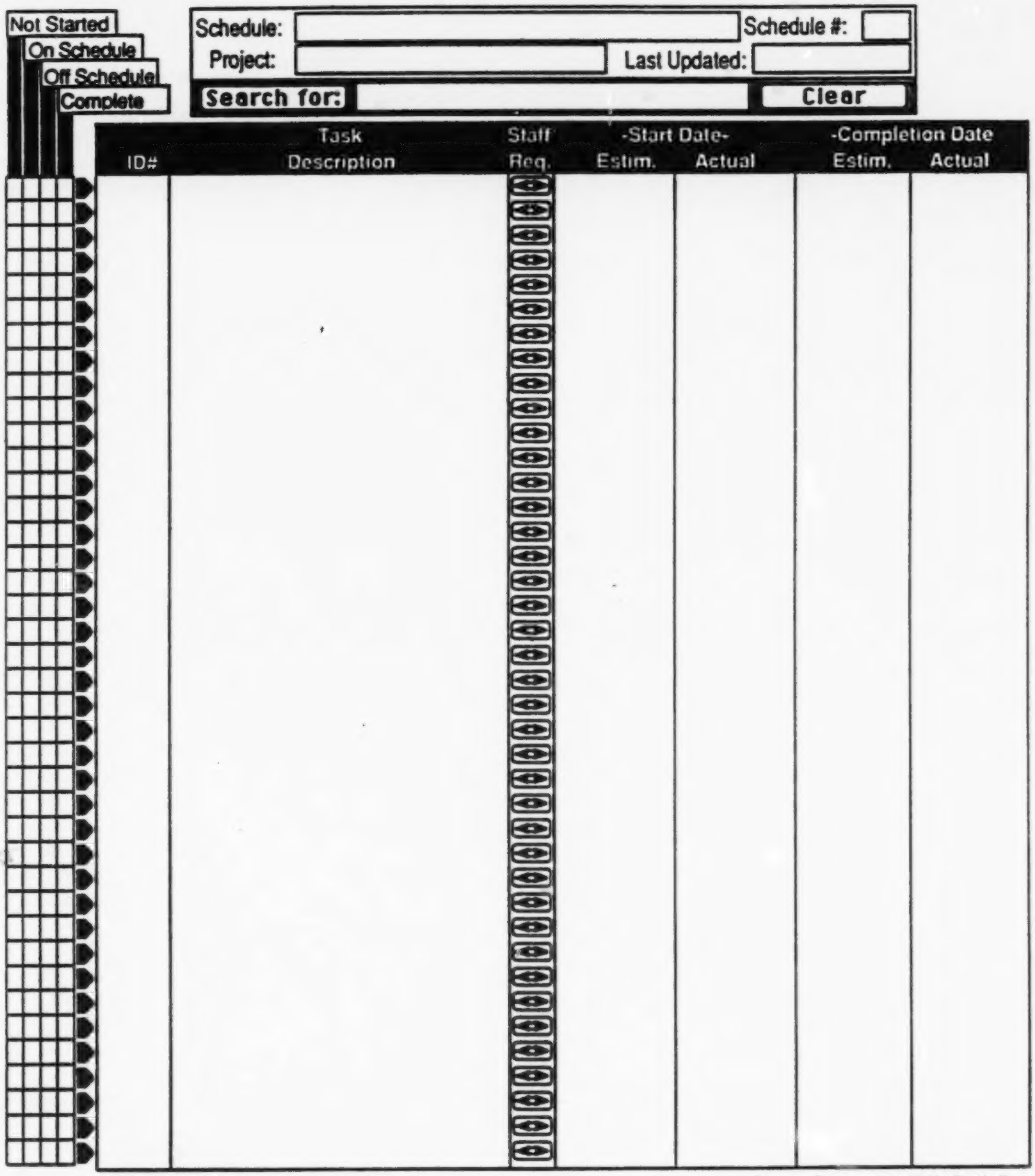
Audio: Video Ch.#1: Srce. tape: Talent:

Video Ch.#2: Srce. tape: Talent:

Script Computer : Srce. disk:

© Interactive Design Managment System 5.1

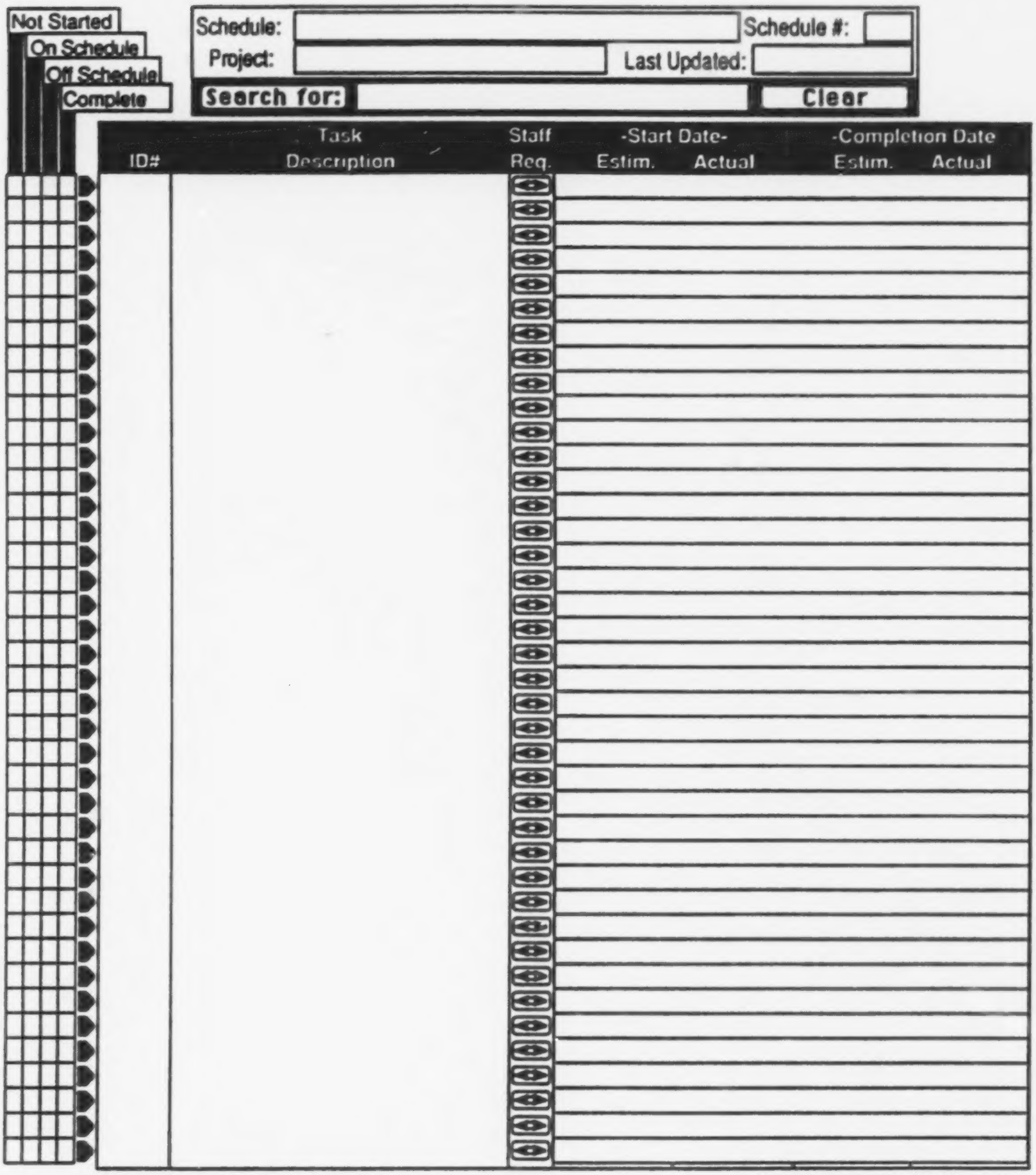
Note: This page has been intentionally left blank to maintain the indexing of the original software manual within this thesis appendix.


Not Started
On Schedule
Off Schedule
Complete
Schedule:
Schedule #:
Project:
Last Updated:
Search for:
Clear
ID#
Task Description
Staff Req.
-Start Date-
Estim.
Actual
-Completion Date
Estim.
Actual

© Interactive Design Managment System 5.1

Synapse Development 89 N. Victoria St.
St.Paul, MN. 55104 (612) 290-0042

Note: This page has been intentionally left blank to maintain the indexing of the original software manual within this thesis appendix.



Not Started
On Schedule
Off Schedule
Complete

Schedule: | Schedule #:
Project: | Last Updated:
Search for: | Clear

ID#	Task Description	Staff Req.	-Start Date- Estim.	Actual	-Completion Date Estim.	Actual

© Interactive Design Managment System 5.1

Synapse Development 89 N. Victoria St.
St.Paul, MN. 55104 (612) 290-0042

Note: This page has been intentionally left blank to maintain the indexing of the original software manual within this thesis appendix.

Index

A
- Accessories menu 10
- Add icon command 32, 33
- Add template page command 42
- Analyis Ideas command 10, 17
- Analysis Outliner 15
  - Analysis Ideas notebook 16
    - chapters 18
    - opening 17
    - turning pages 18
  - opening 16
  - Outline Writer 16, 18
    - changing fonts 20
    - creating files 18
    - deleting files 19
    - exporting text files 19
    - importing text files 19
    - opening existing files 19
    - printing outline files 20
  - parts of 15
- Analysis Outliner command 9
- Arrow keys 18
- Audio output command 42

B
- Back link 9
- Box 11
- Bring front command 33
- Browse tool 11. 33
- Buttons 11

C
- Change icon command 32
- Chart pattern command 32
- Chart size command 32
- Clear template page command 42
- Computer output command 42
- Connector tool 11, 33
- Copy command 8
- Clicking 11
- Current window 11
- Cut command 8

D
- Delete outline file 17
- Delete template page command 42
- Design Templates 34
  - constraining window size 51
  - flowchart relationship 40, 32, 54, 55
  - graphics tools 51
  - headings 49
  - libraries 40, 56
    - creating 56
    - opening 57
  - menus 42
  - opening 40
  - pages 41
    - adding 50
    - branches 52
- Design templates (cont.)
  - pages (cont.)
    - branches (cont.)
      - creating 52
      - deleting 53
      - traveling from 53
      - traveling to 53
    - clearing 50
    - constraining window size 51
    - creating 50
    - deleting 50
    - finds 57
      - field specific 58
      - global 58
    - flowchart icons 54
      - creating 54
      - deleting 55
      - returing from 55
      - viewing 55
    - graphic tools 51
    - headings 4
    - navigating 57
    - printing 61
    - sections of 43
      - audio output 48
      - computer output 47
      - input/branch info 46
      - page info 44
      - video output 48
    - sorting 60
  - preferences 49
  - sorts 60
    - collected 60
    - ordered 60
  - using 39
- DesignTemplates menu 9
- Dialog box 11
- Double clicking 11

E
- Edit menu 8
- Editing text 14
- Element menu 22
- Export text file 17, 19

F
- Fields 11
- Find audio output command 43
- Find computer output command 43

Index

- Flowchart Processor (cont.)
- Find input/branch info. command 43
- Find next command 43
- Find page information command 43
- Find/sort menu 43
- Find videooutput command 43
- First page command 8
- Flowchart processor 21
- command 9
  - icons 33
    - connecting 38
    - creating 33
    - moving 35
    - layers of 38
  - menus 32
  - opening 32
  - pages 36
    - background patterns 33
    - creating 36
    - navigating 37
    - opening 37
    - size 36
  - representative template pages 32
    - creating 33
    - traveling to 36
  - scrolling windows 35
  - tools 33
  - use of 31
- Flowchart toolscommand 32
- Flowcharts menu 32
- Font menu 17, 18, 20
- Frame calculator 10

G

- Global find command 43
- Go menu 9
- Graphics command 42

H

- Hard disk 5
- Hypertool menu 9

I

- Icons 11
- Icon menu 32
- Icon mover tool 11, 33
- IDMS model 22
- Import text file 17, 19
- Input/branching command 42
- Intelligent Design Templates (see Design Templates)
- Introduction 1

L

- Last page command 9

M

- Memory 5
- Menus 7
  - Accessories 10, 63
  - Edit 8, 18
  - Elements 22
  - File 7, 29
  - Find/sort 43
  - Flowcharts 32
  - Font 17, 18
  - Go 9, 37, 57
  - Hypertools 9
  - Icons 32
  - Outliner 17
  - Page 23
  - Templates 32
- Multiple windows 12

N

- New chart command 32
- New outline file command 17
- Next page command 8
- New template library command 42

O

- Open outliner file command 17
- Open template library command 42
- Opening IDMS 6
- Outliner menu 17
- Outline Writer command 10

P

- Page 11
- Page informationcommand 42
- Page menu 23
- Palette 12
- Paste command 8
- Previous page command 8
- Print command 7
- Printing 20
- Procedures 13
- Products 3
- Program launcher command 10
- Project Schedules command 9
- Project Schedules 21
  - adding pages 23
  - calculating 23, 26
  - clearing page 23
  - editing 26
  - hiding staff fields 24
  - inputing dates 26
  - opening 22
  - page parts 24
  - preparing 26
  - printing 29

Index

  showing staff fields 24
  staff responsible functon 26
    assigning 26
    clearing 28
    hiding fields 27
    searching for 27
    showing fields 27
    updating 23, 26
  task linking 28
    creating 28
    deleting 29
    traveling to 29
    viewing path 29
Project preferences 7, 9

Q

Quit command 8

R

Requirements 5
Revert command 8
Running IDMS 6

S

Save command 7, 12
Schedule page parts 24
Script calculator 10
Select all command 32
Send back command 32
Sort by output command 43
Sort by page/input command 43
Speed 12
Staff responsible (see Project Schedules)
Starting a project 13
Sticky notes command 10
System requirements 5

T

Template menu 32
Template preferences command 42
Terminology 10, 12

W

Windows 12

V

Video output command 42

Note: This page has been intentionally left blank to maintain the indexing of the original software manual within this thesis appendix.

Appendix E

Completed Process Evaluation Forms

IDMS Process Evaluation Sheet

Date: 2/20/90 Page # 1 of 4
Evaluator: Grossman

Tool & Task Involved	Can't find necessary information or lost: (Where are you? What's missing?)	Feeling restricted by systems design: (How? Possible solutions?)	Inefficient or inaccurate procedures: (What is? Why? Possible solution?)	Other comments:
1. **Overall System Software:** Installation and operation	Transparent menus and menu options are very user-friendly for experienced IVD personnel, instructive for newcomers.	I appreciated the "I want to use it now!" suggestion in your manual. That's the way most non-computer oriented instructional designers want to learn software capabilities.	No. Menu terms and names very familiar for IVD experienced personnel.	Simple to get up and running! This was my first experience with a Mac. Menus reminded me of Ventura or Publisher's Paintbrush menus--aided in user-friendliness for me. Mouse functions similar.
2. **Project Schedules:** Date setting and calculation	Easy to use.	I felt somewhat limited by the inability to immediately see the impact of scheduling changes on project resource allocations and budet expenitures. It's not a fault of your system, which worked fine, but a limitation that would require another separate software application in large projects such as ours where I had to work within a $1,000,000 budget over 5 years. The kinds of entries possible, handling of vacation days, handling & checking of individual project members' part-time schedules, etc. were features of the more sophisticated Project Management package that would be missed.	Worked fine. In my situation I had overlapping people working on several projects simultaneously. Adjusting their various part-time schedules to many projects would be more difficult with your system than wit HTPM or SuperProject Plus.	I used Harvard Total Project Manager with our first 4 discs, then switched to SuperProject Plus for the last 5 discs because SuperProject Plus was built around true Pert Charts allowing more back-tracking and overlapping of tasks and milestones than HTPM.
3. **Project Schedules:** Setting and searching for staff responsibilities	No problems.		None.	
4. **Project Schedules:** Setting and traveling to task links	Simple to set and use. This was the most useful integrative feature I've discovered to this point. Wish I'd had it available on our projects.		None.	This feature of your integrated system was much more useful than my previous need to shuffle between different software packages to address similar linkages.

IDMS Process Evaluation Sheet

Date: 2/20/90 Page # 2 of 4
Evaluator: Grossman

Tool & Task Involved	Can't find necessary information or lost: (Where are you? What's missing?)	Feeling restricted by systems design: (How? Possible solutions?)	Inefficient or inaccurate procedures: (What is? Why? Possible solution?)	Other comments:
5. **Analysis Outliner:** Writing, saving, and retreiving Outline Writer files	Worked well.	No ability to modify allows for additions, subtractions, etc. as desired.		This tool would have been extremely valuable in the beginning of our work--and would have helped keep us on task during each new project.
6. **Analysis Outliner:** Using the available information	No problems.	Not at all. Open.		We always had weekly meetings of the entire IVD team to keep us on course. This type of tool would have assisted our preparation for those meetings and facilitated their structure.
7. **Flowchart Processor:** Creating and editing a flowchart	Fun and precise.	Only by the time required to move through Level 1, Level 2, and Level 3 flows. System allows everything we've done, but requires more time than paper and pencil.		My designers had a tool like this available, but opted for developing their flow charts on lengthy rolls of butcher block paper tape on walls so that even a complex disc's flow was totally visible at all times.
8. **Flowchart Processor:** Creating and traveling to representative template pages	No problems.	No.		This integrative function is great. Many of our discs took 600+ storyboard pages, though, and I didn't have time to see how that quantity would affect your system.

IDMS Process Evaluation Sheet

Date: 2/20/90 Page # 3 of 4
Evaluator: Grossman

Tool & Task Involved	Can't find necessary information or lost: (Where are you? What's missing?)	Feeling restricted by systems design: (How? Possible solutions?)	Inefficient or inaccurate procedures: (What is? Why? Possible solution?)	Other comments:
9. **Design Templates:** Setting template page preferences and creating new pages	Easy.	No.		We used the 1987 Version 4 of Scripter, a similar template scripting program for IVD developed Cognitive Design Technologies of Coralville, Iowa. It allowed all of the functions of your template designer except
10. **Design Templates:** Recording data on the template pages	Easy to accomplish.	Very open.		the audio feature for narratives allowed by your system and the Mac Scripter also comes complete with videodisc player driver for the pilot testing (if the IVD already exists). Your system templates are easier to use and work with than
11. **Design Templates:** Creating and traveling to representative flowchart icons	An excellent feature.	No.		Scripter's because of the graphics based Mac. They are more esthetically pleasing to work with over long periods. The functions, however, are almost identical. My designer would only use Scripter for 1 disc--preferring paper & pencil
12. **Design Templates:** Creating and pilot-testing template page branching	Very nice and a plus over other design template software I've seen.	No.		storyboarding which they felt more portable so they could work anywhere at any time. They also preferred rough graphics or hand drawn video for the producers or programmers to professionalize.

IDMS Process Evaluation Sheet

Date: 2/20/90 Page # 4 of 4
Evaluator: ______

Tool & Task Involved	Can't find necessary information or lost: (Where are you? What's missing?)	Feeling restricted by systems design: (How? Possible solutions?)	Inefficient or inaccurate procedures: (What is? Why? Possible solution?)	Other comments:
13. **Accessories:** Launching external and returning from programs	Didn't try this tool.			
14. **Accessories:** Calculating script durations and frame numbers	Worked fine.	No.	None.	We used Scripter for this purpose. Yours works as well, although Scripter does adjust to PAL (European codes) not just NTJC.
15. **Accessories:** Creating and deleting notes	Easy.	No.	None.	A nice reminder feature. Would forever be used if my designers had it available. Closure on any disc or segment was our toughest task.
16. **IDMS Model:** Applying the tasks outlined in the model to a design project	All linkages worked as suggested.	No--obviously designed to allow projects as much freedom as desired.		The IDMS model is an excellent integration of the tasks actually engaged in by IVD development groups. Nice job!!

IDMS Process Evaluation Sheet

Date: _________ Page # 1 of 4
Evaluator: Gene Ramsay

Tool & Task Involved	Can't find necessary information or lost: (Where are you? What's missing?)	Feeling restricted by systems design: (How? Possible solutions?)	Inefficient or inaccurate procedures: (What is? Why? Possible solution?)	Other comments:
1. **Overall System Software:** Installation and operation	The addition of more pulldown menu graphics would be helpful for non-Mac users.		Project preferences could have cursor set at project name when first opened.	Easy to install and operate. Text in black boxes does not show up on Mac IIx. (Fonts in the System Folder have been checked).
2. **Project Schedules:** Date setting and calculation				How about pre-formatted Date entry with Tab movement ie. the column look like this when opened:
3. **Project Schedules:** Setting and searching for staff responsibilities		Show/hide staff fields should be an easy to push button/ or power key instead of pulldown menu item.		
4. **Project Schedules:** Setting and traveling to task links		Even with my IIx some of the Link settings are not as fast as I would like it. Maybe a faster XCMD would help.		

BEST COPY AVAILABLE

IDMS Process Evaluation Sheet

Date: ________ Page # 2 of 4
Evaluator: Gene Ramsey

Tool & Task Involved	Can't find necessary information or lost: (Where are you? What's missing?)	Feeling restricted by systems design: (How? Possible solutions?)	Inefficient or inaccurate procedures: (What is? Why? Possible solution?)	Other comments:
5. **Analysis Outliner:** Writing, saving, and retreiving Outline Writer files	Imported text are not saved. Once text files are opened to Analysis Files they cannot be opened again "Data		On Importing text stop "can't find window" came up after transfer.	Stop message comes up when I was in Schedule and choose Analysis Outliner from pulldown menu.
6. **Analysis Outliner:** Using the available information		When opening Outline Files one file said, "Can't find card". Delete File is only for currently opened file.		
7. **Flowchart Processor:** Creating and editing a flowchart		More Icons are needed as a base. Enclosed is a list of additions.		
8. **Flowchart Processor:** Creating and traveling to representative template pages	When creating a representative template page the system is aksing "where is Libary to add to". The software is looking for Libary should be Library			

IDMS Process Evaluation Sheet

Date: ________ Page # 3 of 4
Evaluator: Gene Ramsay

Tool & Task Involved	Can't find necessary information or lost: (Where are you? What's missing?)	Feeling restricted by systems design: (How? Possible solutions?)	Inefficient or inaccurate procedures: (What is? Why? Possible solution?)	Other comments:
9. Design Templates: Setting template page preferences and creating new pages		Duration/Frame fields It would be very helpful is I could use a period instead of a colon in entering timecode from numberkey pad. Else you could have multi-field input with "Tab" access.		
10. Design Templates: Recording data on the template pages		Please allow the user to click "off" the box marked with "x" without clearing all entries.		Warning statements could be helpful, ie. set the size of videodisc to be mastered 8" or 25,200 frames or 12" or 54,000 frames and state if the frames are over or overall project est. layout is on schedule.
11. Design Templates: Creating and traveling to representative flowchart icons	Flowchart icon does not return to template page.			
12. Design Templates: Creating and pilot-testing template page branching				OK

IDMS Process Evaluation Sheet

Date: __________ Page # 4 of 4
Evaluator: Gene Ramsay

Tool & Task Involved	Can't find necessary information or lost: (Where are you? What's missing?)	Feeling restricted by systems design: (How? Possible solutions?)	Inefficient or inaccurate procedures: (What is? Why? Possible solution?)	Other comments:
13. **Accessories:** Launching external and returning from programs		Application names do not always fit in boxes.	If you choose blank icons you get Pagemaker icon.	Upon Return it takes you to beginning of IDMS program not where you left off.
14. **Accessories:** Calculating script durations and frame numbers			Calculator incorrect on Frame Numbers. Redo Formula.	
15. **Accessories:** Creating and deleting notes				Is it possible to move the notes?
16. **IDMS Model:** Applying the tasks outlined in the model to a design project				NA Could not try because of missing or non-functional units.

IDMS Process Evaluation Sheet

Date: 1/22/90 Page # 1 of 4
Evaluator: D. Shapiro

Tool & Task Involved	Can't find necessary information or lost: (Where are you? What's missing?)	Feeling restricted by systems design: (How? Possible solutions?)	Inefficient or inaccurate procedures: (What is? Why? Possible solution?)	Other comments:
1. **Overall System Software:** Installation and operation	Easy to install. I wouldn't call the Macintalk "Macintalk driver (put in system)" because if it's not called "Macintalk", it won't work.	The dedicated system approach is a little weird, but...	You shouldn't let more than one element be open at a time. To confusing Also you can open templates and Analysis, but not Analysis and schedule, so it's weird	Spelled Chad's name wrong in credits--it's "Worrester" Onscreen buttons don't respond to return key--only mouse click
2. **Project Schedules:** Date setting and calculation	Date setting is easy. Calculation seems to work.	Font in actual start date not the same as estimated, so the columns don't match		The date setting lets you put in dates like 1/50/90 and also lets you finish projects before you start them
3. **Project Schedules:** Setting and searching for staff responsibilities	Seem easy. The text starts looking funky when you add names.	let me change the font easily also I don't like how tabbing doesn't scroll	In the documentation, you call the "page" menu the "schedules"menu when discussing how to hide responsible	
4. **Project Schedules:** Setting and traveling to task links	Ooops! From the project management schedule I created a link to an outline writer doc & when I came back, it said "can't send message to closed cards, windows or projects" I cancelled and then the link worked okay, but it closed the document on me.			think the "page" menu should go grey when you don't have schedule open

IDMS Process Evaluation Sheet

Date: 1/22/90 Page # 2 of 4
Evaluator: DAS.

Tool & Task Involved	Can't find necessary information or lost: (Where are you? What's missing?)	Feeling restricted by systems design: (How? Possible solutions?)	Inefficient or inaccurate procedures: (What is? Why? Possible solution?)	Other comments:
5. **Analysis Outliner:** Writing, saving, and retreiving Outline Writer files	I created an outlines tile in project schedules, but couldn't open it in the outline writer "can't find window"	Should be able to resize window to full screen.	I would like to be able to print out the analysis ideas file.	Import is nice & works well.
6. **Analysis Outliner:** Using the available information	started getting pretty buggy restarted under unifinder still couldn't find files I created	You can name 2 outline files the same name		Good to use the book and the outline files.
7. **Flowchart Processor:** Creating and editing a flowchart	icons are excellent. I think the process of creating them is cumbersome though should be able to pick from menus.	Should be able to hit "enter" instead of clicking "okay" buttons	Should be able to change icon name after it's created.	When I flowchart, I do lots of copying and pasting of existing structures, so it would be nice if you could easily copy groups-- I was able to, but you have to be careful.
8. **Flowchart Processor:** Creating and traveling to representative template pages	This seems to work fine. What would be slick is if the links you built turned into lines on the flowchart	It's a great idea, but much too cumbersome for what you get. IF, however, by building the links, you were actually creating the code to control the videodisc, then it would be worth it. As it is, I'd just like and easier way to move through the pages.		

IDMS Process Evaluation Sheet

Date: 1/22/90 Page # 3 of 4
Evaluator: D. Shapiro

Tool & Task Involved	Can't find necessary information or lost: (Where are you? What's missing?)	Feeling restricted by systems design: (How? Possible solutions?)	Inefficient or inaccurate procedures: (What is? Why? Possible solution?)	Other comments:
9. **Design Templates:** Setting template page preferences and creating new pages	Seems easy.	Would like to be able to see more of the page.	The template seems to reflect a CBT approach less than videodisc not much space for the actor's lines.	Find works okay, but it should create a group with those criteria, not just find what works
10. **Design Templates:** Recording data on the template pages	Works okay I hate scrolling sideways.	Tab order is weird		Should be able to generate reports.
11. **Design Templates:** Creating and traveling to representative flowchart icons		You can't go backwards to an existing flowchart When I click the flowchart box it lets m link, bu I want to link it to an already existing chart and I can't		
12. **Design Templates:** Creating and pilot-testing template page branching	This works okay, but its too much work, I think for what you get out of it.			

IDMS Process Evaluation Sheet

Date: 1/22/90 Page # 4 of 4
Evaluator: D. Shapiro

Tool & Task Involved	Can't find necessary information or lost: (Where are you? What's missing?)	Feeling restricted by systems design: (How? Possible solutions?)	Inefficient or inaccurate procedures: (What is? Why? Possible solution?)	Other comments:
13. **Accessories:** Launching external and returning from programs		Would like to be able to pick icons	Would be nice if it broughtyou back to where you exited.	
14. **Accessories:** Calculating script durations and frame numbers		The calculations should happen on the script by putting in times there.		I love the script script calculator. Duration calculator should not have hours. Duration calculator still dies @ about 00:0:18
15. **Accessories:** Creating and deleting notes	Nice except you should be able to move the notes around			
16. **IDMS Model:** Applying the tasks outlined in the model to a design project		Overall, by sense is that this IS how you have to look @ the project, but I'm not sure filling all this out will be worth all the effort other than it will make you do it STILL, overall this is excellent, exciting & fun!!		

Appendix F

Completed Summative Evaluation Forms

Date: 2/20/90
Evaluator: Grossman

IDMS Overall Evaluation Sheet

Yes	?	No		Comment
X			1. Will the system increase work efficiency?	The structure and task focus provided by the system would focus integrated activities.
X			2. Will the system increase work accuracy?	My designers have balked when requested to use computerized flowcharting and storyboarding aids in the past. If they would, accuracy would increase.
	X		3. Is the system capable of working within the parameters of your current work procedure?	The task parameters, yes. The people parameters, no. The multiple disc projects with part-time people scheduled simultaniously to different projects, maybe.
X			4. Is the IDMS model complete?	An excellent integration of tasks and tools in an integrated package. The varied communication, supply, training and morale needs are absent.
	X		5. Is the system complete?	For large projects, more project management tools for resource and budget allocations would be required. For small projects, yes, except for items noted above.
		X	6. Does the system offer unique design and pilot-testing capabilities?	The integration of the simplier tools is unique. However, Scripter 4.0 offers the same instructional design and pilot-testing capabilities without the other components.
X			7. Would students learning the IV development process benefit from using this system?	I believe that this is your system's greatest asset! As a teaching tool, it introduces the complexities of IVD development in an understandable, structured, and flexible manner.

8. Which tools and accessories did you use? How useful was each tool?

Used?			Usefulness			
Yes	No		Very	Some	None	Comment
		Hypertools				
X		Project Schedules		X		Too many other variables (ie. budget, vac., part-time schedules, simult. projects) need to be factored in.
X		Analysis Outliner	X			An excellent lit. review product.
X		Flowchart Processor		X		My designers prefer to do it manually.
X		Design Templates	X			An excellent storyboarding tool, but will be difficult to get creative, non-computer oriented designers to use.
		Accessories				
	X	Launcher				
X		Outline Writer	X			Handy and easy to use.
X		Analysis Ideas	X			An excellent guideline, & modifiable.
X		Sticky Notes	X			Designers would use frequently.
X		Frame Calculator	X			Useful, same as Scripter, add PAL?
X		Script Calculator	X			First time I've seen.

9. What is the system's greatest strength?

An integrated set of computer-based IVD development tools based on an accurate model of the complex IVD development process in one package. Would be an inexpensive system for teaching hands-on IVD development in courses and would work well with computer-friendly personnel working on relatively small projects.

10. What is the system's greatest weakness?

The system lacks many of the more sophisticated project management, statistical analysis, budget allocation, and spreadsheet tools required in large IVD projects with multiple-disc projects developing simultaneously and staffed by a cadre of part-time personnel.

Date: 2/23/90

IDMS Overall Evaluation Sheet

Evaluator: Ramsay

Yes	?	No		Comment
X			1. Will the system increase work efficiency?	No stone is unturned.
X			2. Will the system increase work accuracy?	
	X		3. Is the system capable of working within the parameters of your current work procedure?	Most analysis and scheduling tools are not needed.
		X	4. Is the IDMS model complete?	Prototype applications should be able to be connected to a videodisc player for testing.
	X		5. Is the system complete?	It's complete to where it currently ends. More icon options needed for flowcharts.
X			6. Does the system offer unique design and pilot-testing capabilities?	
X			7. Would students learning the IV development process benefit from using this system?	Yes!!

8. Which tools and accessories did you use? How useful was each tool?

Used?			Usefulness			
Yes	No		Very	Some	None	Comment
		Hypertools				
X		Project Schedules		X		
X		Analysis Outliner		X		
X		Flowchart Processor	X			Great item. Needs more icons.
X		Design Templates		X		
		Accessories				
X		Launcher	X			
X		Outline Writer		X		
X		Analysis Ideas		X		
X		Sticky Notes	X			
X		Frame Calculator	X			
X		Script Calculator		X		

9. What is the system's greatest strength?

For teaching interactive design.

10. What is the system's greatest weakness?

More information thanis needed for normal projects.

Date: 1/30/90
Evaluator: Shapiro

IDMS Overall Evaluation Sheet

Yes	?	No		Comment
	X		1. Will the system increase work efficiency?	Hard to say. There's a lot involved in using the system. Might take too much time.
X			2. Will the system increase work accuracy?	
		X	3. Is the system capable of working within the parameters of your current work procedure?	Our oranization is not big enough to require this level of reporting.
X			4. Is the IDMS model complete?	
X			5. Is the system complete?	
X			6. Does the system offer unique design and pilot-testing capabilities?	I think they exist in other places, although not in one package.
X			7. Would students learning the IV development process benefit from using this system?	Yes, definitely.

8. Which tools and accessories did you use? How useful was each tool?

Used?			Usefulness			
Yes	No		Very	Some	None	Comment
		Hypertools				
X		Project Schedules	X			Finding who does what seems useful. Schedule updating useful. Links not very useful.
X		Analysis Outliner		X		Memory jog of ideas is best feature.
X		Flowchart Processor		X		The flowchart is pretty good, linking is good feature, but too cumbersome for what you get.
X		Design Templates		X		
		Accessories				
X		Launcher		X		Other products do it better.
X		Outline Writer		X		I'd rather use some other text editor .
X		Analysis Ideas	X			This is really good.
X		Sticky Notes		X		Should be able to move notes.
		Frame Calculator				
		Script Calculator				

9. What is the system's greatest strength?

The importance is that it forces you to consider all the necessary elements.
The documentation is excellent.

10. What is the system's greatest weakness?

I think, like the videodisc process, it's necessary to pull from a variety of sources and create a custom design process. The integrated nature of IDMS makes it rather unwieldy for my needs.

Appendix G

Software Distribution Information

IDMS Software Distribution

Information on the distribution of the Interactive Design Management System software can be obtained by contacting Project MinneMac.

Project MinneMac
132 Shepherd Labs
100 Union Street SE
Minneapolis, Minnesota 55455
(612) 625-0141